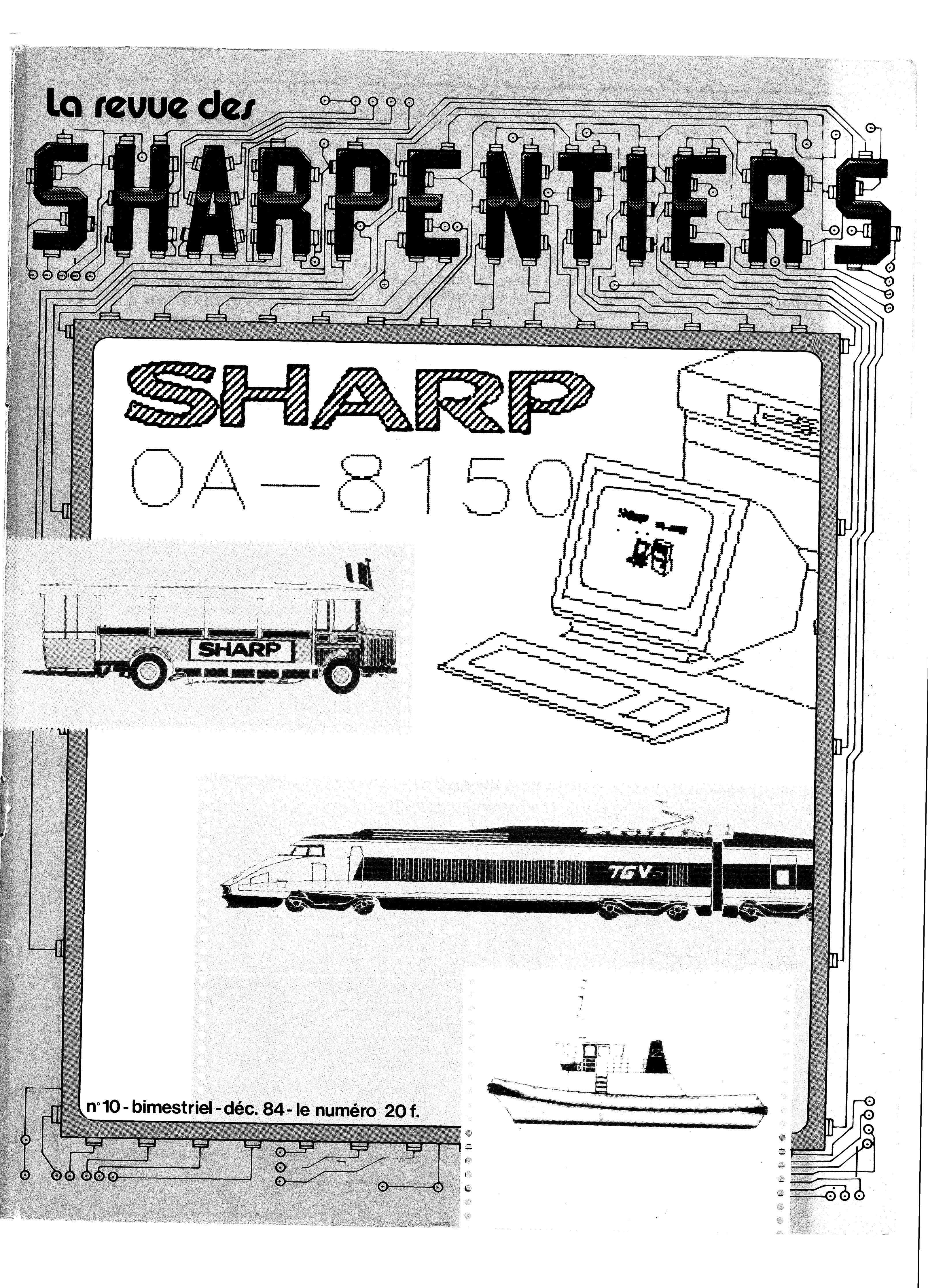
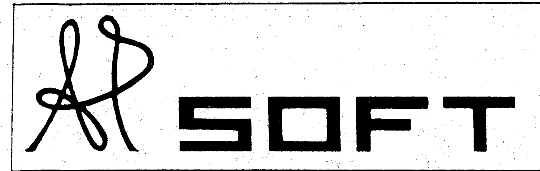


La revue des
SHARPENTIERS
SHARP
OA-8150
SHARP
TGV
n°10 - bimestriel - déc. 84 - le numéro 20 f.



LE SECOND SOUFFLE DE VOTRE POCKET

LISTE DES LOGICIELS DISPONIBLES POUR PC 1500

SOFT CALC	: tableur professionnel avec graphiques
SOFT GRAPH	: graphiques de gestions et scientifiques
SOFT STAT	: traitement statistique des données avec graphiques
SOFT TEXT	: traitement de textes sur CE 150 et imprimante paralléle
SOFT DRAW	: représentation graphique en trois dimensions
SOFT FONCTION	: tracé de courbes et de surfaces
SOFT EXTEND	: extension de 70 fonctions au BASIC
SOFT MONITOR	: moniteur-désassembleur en mnémoniques officielles
SOFT SYSTEM	: gestion d'une partie de la RAM en disquette (RAM-DISK)
SOFT MIND	: 3 jeux utilisant les principes de l'intelligence artificielle
SOFT GAME 1	: 6 jeux video en langage machine
SOFT GAME 2	: 4 jeux video en langage machine

BIENTOT POUR VOTRE PC 1350

FREE CALC	: tableur professionnel (feuillle de calcul)
FREE TEXT	: traitement de textes sur CE 126 et imprimante série
FREE STAT	: traitement statistique des données
FREE GRAPH	: graphiques de gestion et scientifiques (sur CE 515 P)
FREE BASE	: gestion de fichiers avec possibilité de calcul.
GAME 1	: jeux video
GAME 2	: jeux video
GAME 3	: jeux de dés et de cartes
GAME 4	: jeux de reflexion
GAME 5	: jeux utilisant les principes de l'intelligence artificielle

Tous nos logiciels sont livrés sur cassette AUDIO . avec une notice explicative detaillée . Ils sont disponibles chez votre revendeur SHARP , au club des SHARPENTIERS et par correspondance .

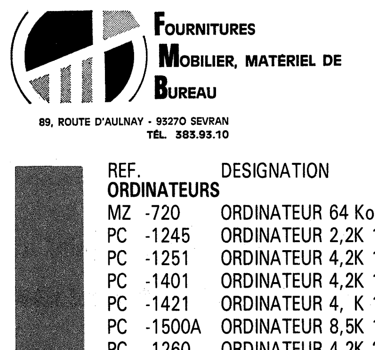
Pour plus de RENSEIGNEMENTS contactez chez AP SOFT :
Mr Pascal ABRIVARD

AP SOFT
1 bis rue du printemps
78230 Le PECQ
tel.: (16) 976 83 74

AP SOFT se reserve le droit de faire toute modification sur les caractéristiques annoncées et cela sans préavis.

Vous pouvez également vous procurer les logiciels de la Société A.P. SOFT auprès du CLUB DES SHARPENTIERS aux tarifs ci-dessus (port compris).

PROFESSIONNELS,
REVENDEURS
CETTE SURFACE
VOUS EST CONSACREE



FOURNITURES MOBILIER, MATÉRIEL DE BUREAU

89, ROUTE D'AULNAY - 93270 SEVRAN
TÉL. 383.93.10

REF.	DESIGNATION
ORDINATEURS	
MZ -720	ORDINATEUR 64 Ko
PC -1245	ORDINATEUR 2,2K 1L.16C
PC -1251	ORDINATEUR 4,2K 1L.24C
PC -1401	ORDINATEUR 4,2K 1L.16C
PC -1421	ORDINATEUR 4, K 1L.16C
PC -1500A	ORDINATEUR 8,5K 1L.26C
PC -1260	ORDINATEUR 4,2K 2L.24C
PC -1261	ORDINATEUR 10 K 2L.24C
PC -1350	ORDINATEUR 5K 4L.24C
IMPRIMANTES / LES INTERFACES	
MINI-1P01	INTERFACE IMPRIMANTE
I.T.	IMPRIMANTE HR5
I.N.	IMPRIMANTE M1009
LES INTERFACES	
CE -124	INTERFACE CASSETTE 245/251/401/421/260/261
CE 126P	IMPRIMANTE INT. CAS. 245/251/401/421/260/261
CE -125	MAGNETOPHONE M.C.I. 1245/1251/1260/1261
CE -150	IMPRIMANTE INT. CAS.
LES EXTENSIONS ET OPTIONS PC 1350	
CE -201M	EXTENSION RAM 8K P.
CE -202M	EXTENSION RAM 16K P.
CE -515P	IMPRIMANTE G. 4C 80C
CE -516L	CABLE DE LIAISON
LES EXTENSIONS ET OPTIONS PC 1500A	
CE -152	MAGNETOPHONE MICRO CAS
CE -153	CLAVIER SENSITIF
CE -155	EXTENSION RAM B K
CE -158	INTERFACE RS 232C/CENT
EA -158	CABLE (PARALLELE CENT.)
CE -159	EXTENSION RAM 8 K P.
CE -161	EXTENSION RAM 16 K P.
LES EXTENSIONS ET OPTIONS PC 1500A	
CE -152	MAGNETOPHONE MICRO CAS
CE -153	CLAVIER SENSITIF
CE -155	EXTENSION RAM 8K
CE -158	INTERFACE RS 232C/CENT
EA -158C	CABLE (PARALLELE CENT.)
CE -159	EXTENSION RAM 8K P.
CE -161	EXTENSION RAM 16 K P.
LES EXTENSIONS MZ 720	
MZ -1D05	ECRAN COULEUR
MA 1X03	MANETTE DE JEUX
SFD	SIMPLE LECTEUR DISQUETTES
DFD	DOUBLE LECTEUR
C80	CARTE 80 COLONNES
CPM 2.2	DISQUETTE PLUS MANUEL
I.C.	INTERFACE CENTRO
LES FOURNITURES	
R.700	ROULEAUX TUBE DE 6
850C.B	CRAYON LES 4C OU 4N
D.DF.DD	DISQUETTE 5"PAR 10
D.DF.DD	DISQUETTE 5"PAR 100
D.N	DISQUETTES NETTOYAGE

SOMMAIRE N° 10

MAQUETTE ET FABRICATION IN-QUARTO. 19, RUE F.LEMAITRE 75020 PARIS. TEL. 636.73.04



Tradition oblige, le nouvel an, période faste en promesses et bonnes résolutions, nous conduit tout naturellement à vous faire part des nôtres.
Côté promesses : vous recevrez, en 1985, 6 numéros du SHARPENTIER dont 1 numéro « SPECIAL VACANCES - SPECIAL PROGRAMMES ».
Nous nous efforcerons également de raccourcir le délai de réponse au courrier qui est actuellement dans la grande majorité des cas, inférieur à 15 jours.
Côté bonnes résolutions : nous vous soumettrons dans le N° 11, un questionnaire/sondage qui, après analyse, nous permettra de mieux connaître votre appréciation sur les services que nous vous proposons, et de mieux faire ''coller'' le contenu de nos futurs bulletins ainsi que l'ensemble de nos activités à vos aspirations et à vos souhaits.
Puisque nous parlons résolution, admirez celle de l'imprimante SHARP IO 700 qui a créé les dessins de notre couverture : 1024 points par ligne, en 8 couleurs, (la couleur, c'est pour bientôt dans le SHARPENTIER). Connectée au ''petit dernier'' de SHARP : L'OA 8100 qui n'a de petit que le nom. Jugez-en plutôt : 4 méga octets de RAM, 270 M.O. sur disque dur, microprocesseur 68 000, et capable d'effectuer 32 tâches simultanées... Ne rêvons pas, ... à nos micros... et à bientôt.

S. BIZOIRRE



REDACTEUR EN CHEF
S. BIZOIRRE
REDACTEUR EN CHEF ADJOINT
L. BURELLIER
PUBLICITE
S. BIZOIRRE/834.93.44

ONT PARTICIPE A CE NUMERO
P. ABRIVARD, A. BERMOND, C. CAMUS, S. CHAGNOUX, V. CREUX, B. EDOUARD, N. GILLES, M. GIRONDOT, J.P. GLUCK, P. HENRY, B. KOKANOSKI, W. KOMBAR, J.F. LENTE, M. LOUTTER, G. NICOLAS, V. OSTROMOURHAON, M. STERMANN, J.F. VIGNAUD, G. VINCENT

LA REVUE DES SHARPENTIERS

LE MZ 5600

L'annonce d'un nouveau produit créé par SHARP finit par ne plus nous étonner, serions-nous blasés ?
Il est vrai que tous les SICOB'S apportent leurs pesant de PC, toujours plus puissants et toujours plus petits. Moins souvent, mais tout aussi spectaculairement, l'on voit surgir un MZ et parfois même si l'on sait bien ouvrir les yeux, un OA. Les PC ont tous des microprocesseurs 8 bits CMOS spécifiques SHARP sauf dans le SPC-5000 ou le I-8088 sert à la compatibilité IBM, mais vous le trouverez comme esclave d'un 8 bits hyper-puissant qui adresse plus d'un Méga octet de mémoire.
Les OA reposent sur le MC-68000, le fameux 16/32 bits de Motorola, et tournent sous le système d'exploitation multipostes multi-utilisateurs multitaches Unix. Ce dernier a toutes les chances de devenir le système d'exploitation standard des années 85 à 90 du fait de la simplicité et l'homogénéité de sa conception. Le SPC-5000 dispose aussi d'une des formes d'Unix : VENIX qui fonctionne avec disquettes et est à l'étude au Japon sur ROM 512 KO. Tout cela ne nous étonne même plus !
Pourtant, le nouveau MZ-5600 devrait bien retenir votre attention pour plusieurs raisons : un vrai processeur 16 bits intel-8086, associé à sa batterue de coprocesseurs, le 8087 (en option) : processeur arithmétiques, le 8288 contrôleur de bus et les PICS gérant les priorités d'interruptions ; 256 à 512 Koctets de mémoire centrale plus 96 à 1932 Koctets de mémoire graphique (3 ou 6 pages de 640 × 400 points) ; un PIO programmable : le 8255 qui a pour tâche de gérer l'interface parallèle CENTRONICS en sortie, et le processeur CMOS du clavier en entrée ; un SIO : le Z80-SIO qui gère un port série synchrone ou assynchrone et un second uniquement assynchrone ; la mémoire graphique est adressée directement par le processeur 8086 en mots de 16 bits, et partagée entre ce dernier et, d'une part le processeur graphique 7220, et d'autre part, par 2 processeurs programmables exclusivement dédiés à la gestion des fenêtrages à l'écran et des priorités de couleurs, et disposant de leur propre mémoire CMOS ; une horloge date/heure disposant d'une d'accus au NIC.D ainsi que d'un générateur de son, qui, non content de vocaliser sur 8 octaves / 3 voies, peut gérer le multiplexage de bancs de mémoire de 128ko à 1 Mo inutilisé à l'heure actuelle, mais utile pour un disque RAM.

Tous ces avantages matériels ne seraient rien si le système d'exploitation ne savait en faire le meilleur usage.
EOS 16 (European, Operating System 16 bits), est un système dont la structure en couche est très largement inspirée d'UNIX, et écrit en langage C. C'est un système multitâche disposant de dix terminaux virtuels dont deux sont visualisables à l'écran simultanément compatibles à 100% avec les concurrents CP/M, CPM-86 et grâce au filtre, compatible MS-DOS 1.4 et 2.0 et même, si l'on dispose de la carte Z-80, compatible EOS V3 (8 bits) et CP/M-80. « Comment », me direz-vous, cela peut-il fonctionner ?
— « Oh, bien, le plus simplement du monde ! »
— vous répondrai-je, « Il n'y a même pas à faire attention ».
EOS reconnaît tout seul que tel ou tel fichier est au format MS-DOS ou CP/M et le traite en conséquence. Moralité, vous pouvez lancer un multiplan sous CP/M80 un WORDSTAR sous MS-DOS et une DBASE II sous CP/M 86 simultanément, et les copier sur la même disquette ou disque dur (eh, oui !) sans vous poser la moindre question.

Utilisant des disquettes double face, double densité, plusieurs formats sont possibles : en 80 pistes : 640 Ko (compatibilité IBM PC oblige), et 800 Ko (compatibilité MZ-35/3), en 40 pistes : le 320 Ko (SPC-5000 et IBM-PC). Le langage de programmation est le C de DESMET, qui permet la programmation « ciselée » à tous les niveaux et une portabilité comme jamais il n'y en a eu : depuis les 8 bits jusqu'aux grands minis sous UNIX. Un grand nombre d'appels système sont disponibles pour le contrôle des processus, (ou tâche) ainsi que les lectures/écritures disque et disquettes.
Le système EOS 16 dispose d'un interpréteur de commandes : le SHELL qui est presque aussi puissant que celui d'UNIX.
Redirection des entrées/sorties, « pipes » ou « tubes », /Dev le répertoire des périphériques, mount, un mount, etc. sont présents. Gageons, que très rapidement, un grand nombre de personnes développeront des utilitaires pur cette machine.
Rendez-vous pour plus d'informations dans le prochain numéro.

Bernard EDOUARD

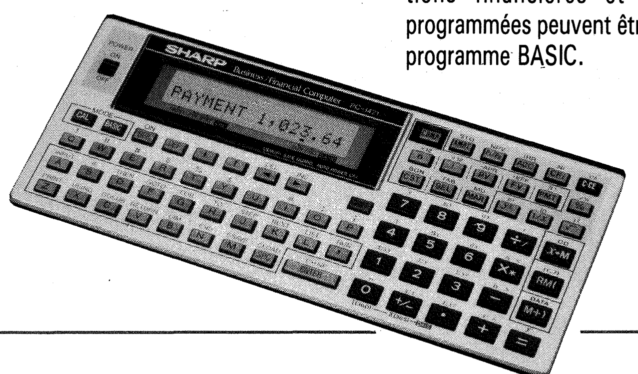


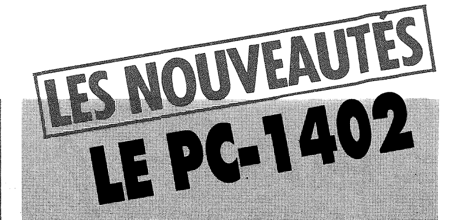
LES NOUVEAUTÉS
LE PC-1421

Mettez côte à côte, un scientifique et un homme d'affaires, faites de même avec un PC-1401 et un PC-1421. Entre les deux hommes, très peu de différence, rien ne ressemble plus à un homo sapiens qu'un autre homo sapiens ; idem pour nos deux PC. Même taille, même afficheur, même disposition des touches. La tête de nos deux hommes est bourrée de chiffres, mais chacun les manipule à sa manière : l'un pense factorielle, hypoténeuse, réciproque, l'autre investissement, statistiques, rentabilité. Le PC-1401, après plusieurs mois



de carrière est maintenant universellement reconnu comme étant l'outil indispensable, de notre scientifique. Notre homme d'affaires aura maintenant à sa disposition un outil tout aussi puissant spécialisé dans les études financières, grâce à ses fonctions pré-programmées, également utilisables en BASIC, le PC-1421, peut calculer immédiatement, le cours d'une monnaie, un taux d'intérêts composés, un amortissement une marge bénéficiaire, effectuer des calculs prévisionnels, faire des statistiques, bref tous les calculs financiers utiles à notre banquier ou chef d'entreprise. Outre ses fonctions scientifiques, le PC-1421 possède la même taille mémoire que le PC-1401 (3,5 Kg), et le même BASIC. Rappelons que toutes les fonctions financières et statistiques pré-programmées peuvent être intégrées dans un programme BASIC.





Le grand frère du PC-1401. Un air de famille indéniable : même présentation sauf un petit liseré rouge au dessus de l'afficheur, même basic, mêmes fonctions scientifiques, même disposition des touches. La différence ? 6,5 ko ; puisque le PC-1402 dispose maintenant de 10 Ko de mémoire, vive disponible aux programmes de l'utilisateur, de quoi combler les plus exigeants.

Nota :
Le PC-1401, ne peut pas bénéficier de cette mémoire additionnelle).

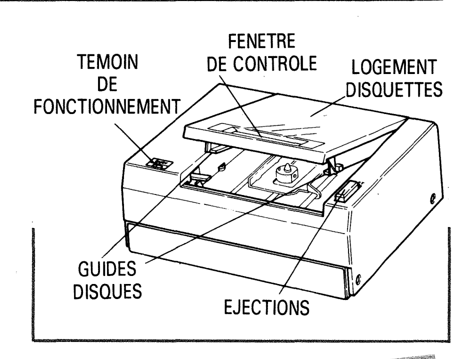


DISQUETTES 5 POUCES

Cette unité de disquettes, décrite dans notre N° 9 de septembre 84, existe maintenant en double unité ; logées dans un même boîtier permettant la recopie directe d'une disquette sur une autre et facilitent la gestion de fichiers importants (560 K formatés disponibles).
Elles sont en outre parfaitement adaptées au fonctionnement du MZ-700 sous système d'explotation C-PM*.

-QUICK DISC-

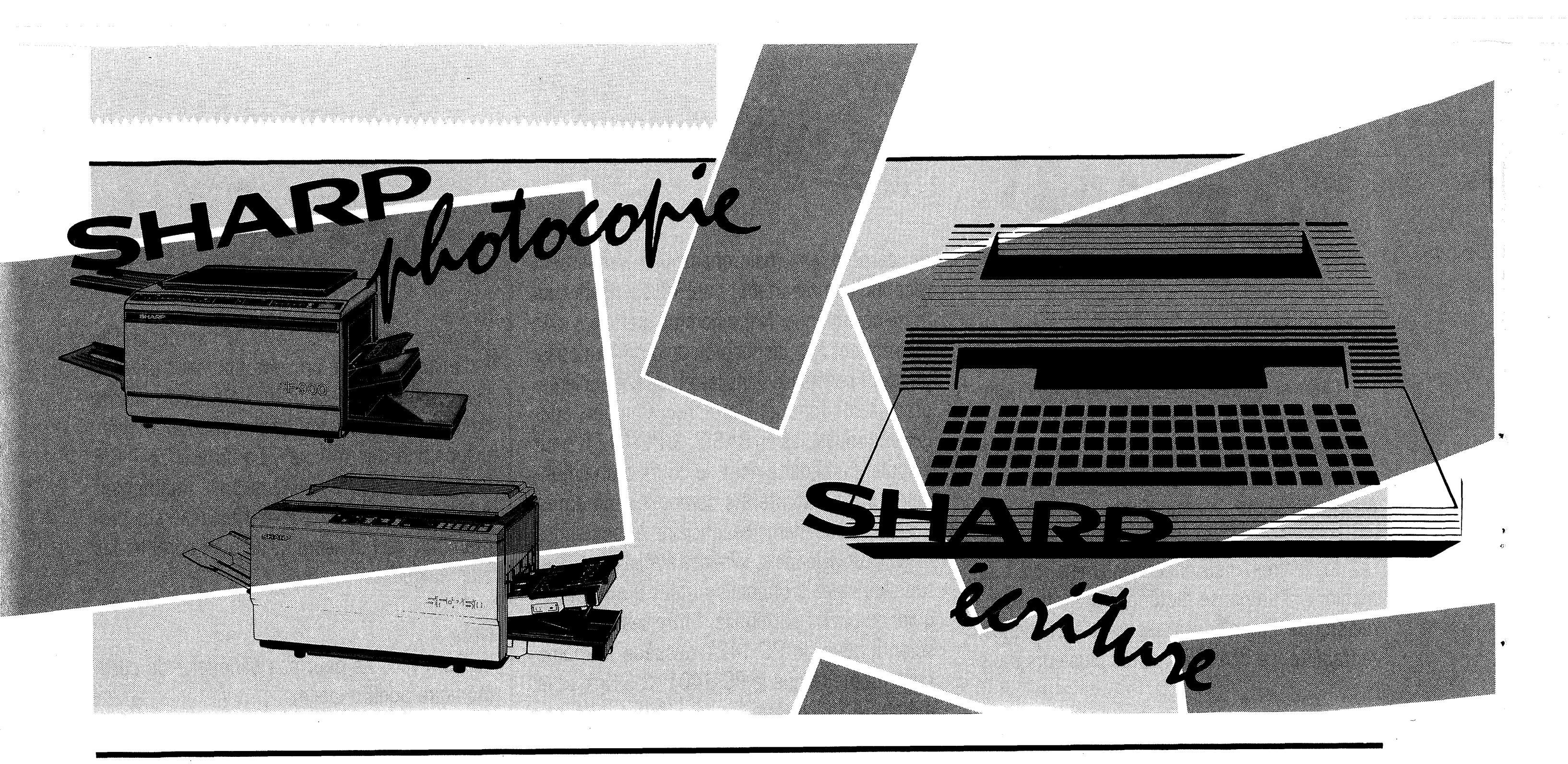
Votre consommation de café devient vraiment très importante lorsque vous attendez le chargement de vos programmes, votre gentil frère a enregistré sa musique préférée sur votre dernière cassette de programmes... Rassurez vous tout cela appartient maintenant au passé, le QUICK DISC est arrivé. Qu'est-ce que le QUICK DISC ? - C'est une unité de disquettes 2,8 pouces qui se loge en lieu et place de votre magnétophone et qui fait rigoureusement la même chose... 60 fois plus vite. Le chargement du BASIC se fait en 3,5 secondes (un rêve !), et l'accès à un programme quelconque prend 8 secondes maximum. Les disquettes utilisées ont une capacité de 64 K formatés (utilisables), par face, donc 128 K en tout. Vous pourrez enregistrer, lire, effacer vos programmes, lister le catalogue des programmes, ou fichiers contenus sur votre disquette ou transférer vos programmes de la cassette vers la disquette le magnétophone conservant son entière autonomie.



CARTE 80 COLONNES

Cette carte se connecte sur le BUS de votre MZ et permet l'affichage de 2000 caractères (80 × 25) sur un moniteur vidéo noir et blanc. Le connecteur du MZ étant monopoliséà par cette carte, une seconde sortie est prévue pour la connection éventuelle de disquettes 5 pouces, l'ensemble CARTE 80 COLONNES + DISQUETTES 5 POUCES + MONITEUR NOIR ET BLANC rend votre MZ 700 compatible à 100% avec le système d'exploitation C-PM * et à sa vaste bibliothèque de logiciels.

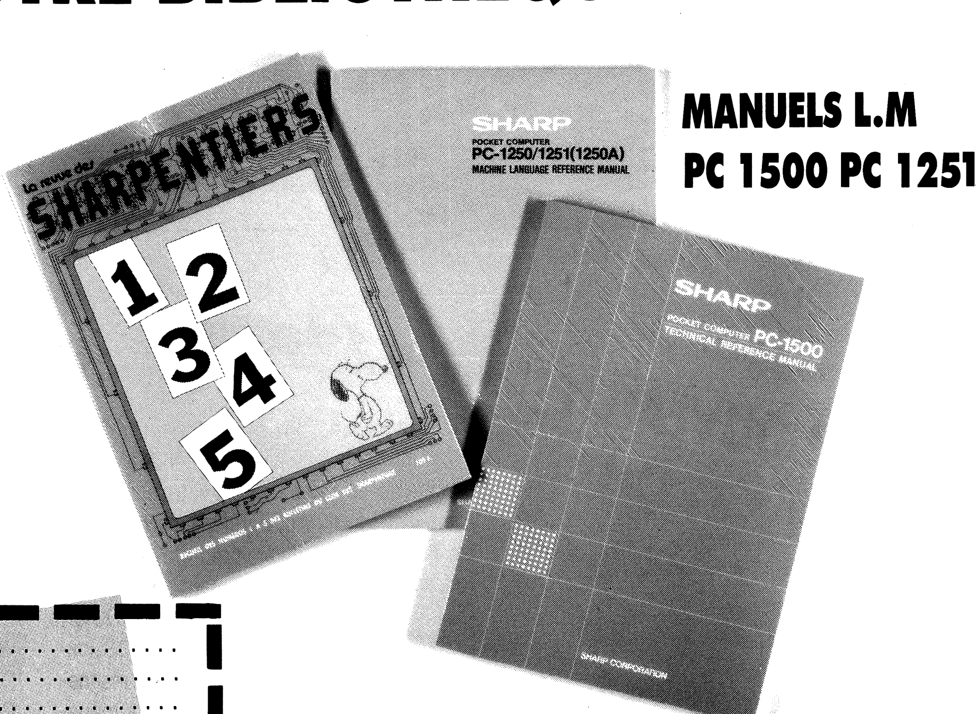
** C-PM est une marque déposée par DIGITAL RESEARCH.*



DANS VOTRE BIBLIOTHEQUE...

LE SHARPENTIER

Entre PROUST et VOLTAIRE, vous pourrez maintenant glisser votre volume du SHARPENTIER. Nous avons rassemblé, en un seul volume de 120 pages, la reproduction fidèle des numéros 1 à 5 du bulletin du club. Ce recueil vous fera participer à la naissance de notre club (1981) et vous fournira tous les programmes, astuces et connaissances acquis durant les 2 premières années d'activité des Sharpentiers. Bien au-delà, l'achat de ce volume représente un investissement durable que les bibliophiles du monde entier s'arracheront dans les siècles à venir...



MANUELS L.M
PC 1500 PC 1251

M. ....
ADRESSE ....
....
vous prie de bien vouloir lui faire parvenir
☐ Recueil(s) des n° 1 à 5 des bulletins SHARPENTIERS. au prix de 110 F. (port compris)
☐ Bulletin(s) SHARPENTIER n° 6
☐ Bulletin(s) SHARPENTIER n° 7 AU PRIX DE 20 F. CHACUN
☐ Bulletin(s) SHARPENTIER n° 8
☐ Bulletin(s) SHARPENTIER n° 9
☐ Manuel(s) Langage Machine PC 1500
☐ Manuel(s) Langage Machine PC 1251
au prix de 167,40 F. l'un (port compris).
Je sais que ce livre est en anglais
et qu'il s'agit d'un manuel de référence
Ci-joint un chèque de .... francs.
Date Signature

Disponibles en exclusivité au club des Sharpentiers, vous pouvez vous procurer ces 2 manuels soit par correspondance au prix de 167,40 F. l'un (port compris) soit directement au club, le mercredi après-midi (exclusivement) au prix de 150 F.

Ces manuels, en anglais très bien documentés, sont des ouvrages de référence et n'abordent par l'initiation au langage machine.



CONCOURS INTERNATIONAL D'OTHELLO L'O.I./SICOB 84

CATEGORIE POCKET

1. WAËL KOMBAR PC 1500
2. CHRISTIAN CAMUS PC 1500
3. M. CAPRON PC 1500

Après nos victoires au TOURNOI INTERNATIONAL, en exclusivité pour les SHARPENTIERS, nous vous proposons l'explication détaillée de nos deux programmes. Ces renseignements, nous l'espérons, permettront aux néophytes de prendre goût aux jeux de réflexion, et offriront peut-être aux plus expérimentés d'entre vous une nouvelle approche de ce jeu ; et pourquoi, par la même occasion, ne pas appliquer ces explications à la programmation du jeu de dames, objet du grand concours du club. 1 semaine au Japon... Ça en vaut la peine !!.

Tout programme de jeu de réflexion, comporte deux parties principales distinctes : la première est le « générateur de coups » consistant, comme son nom l'indique, à générer et simuler tous les coups possibles que peut jouer l'ordinateur. La deuxième grande phase, « l'évaluation » permettra de classer par ordre de valeur l'ensemble des coups jouables, afin que l'ordinateur puisse jouer celui qui lui paraît le meilleur : Etudions successivement les deux phases du programme.

1ERE PHASE : LA GENERATION DES COUPS

Cette étude de tous les coups jouables a été jusqu'alors, la principale méthode utilisée par les programmes OTHELLO BASIC. Cependant, évaluer les 6 ou 7 coupes immédiatement jouables par l'ordinateur ne prend en langage machine que quelques centièmes de secondes, (l'ordinateur dispose en concours, d'une heure pour jouer sa partie !). Ainsi a-t-on pu penser qu'une recherche plus approfondie pouvait être effectuée. A chaque coup jouable par l'ordinateur, on pourrait envisager toutes les parades possibles de l'adversaire, puis de même, à chacune de ces parades chercher toutes les ripostes offertes à l'ordinateur, et ainsi de suite..

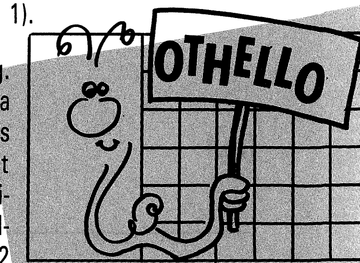
Cette recherche en profondeur, « à plusieurs demi-coups », permet de prévoir les différentes évolutions de la partie dans un avenir plus ou moins éloigné.

Ainsi forme-t-on ce que l'on appelle un « arbre de recherche ». VOIR SCHEMAS.

Cependant, l'évaluation ne donne les notes qu'aux différents coups du bout de l'arbre ; il s'agit alors de « remonter » l'arbre pour sélectionner le meilleur coup.

Attention, il serait ridicule de garder la meilleure des notes calculées par l'évaluation sur l'ensemble du dernier demi-coup !

Un petit exemple vous le fera comprendre rapidement (voir fig. 1).

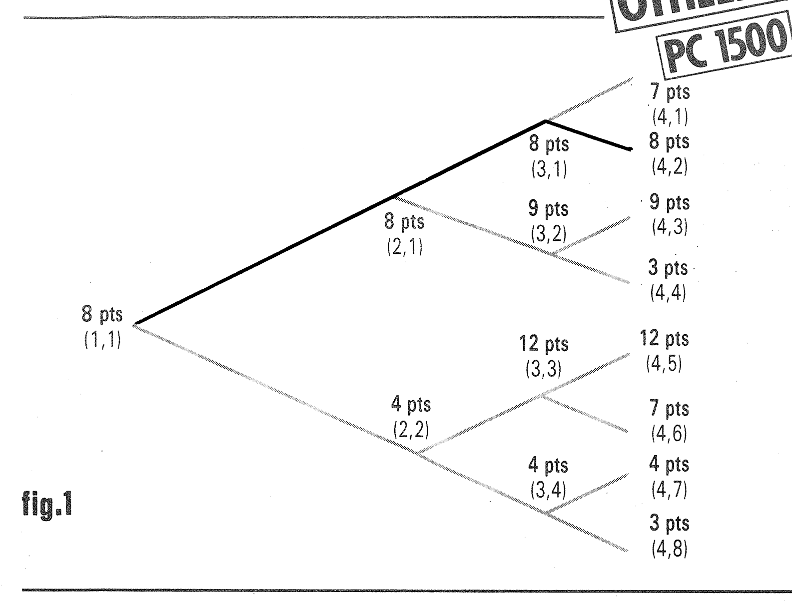


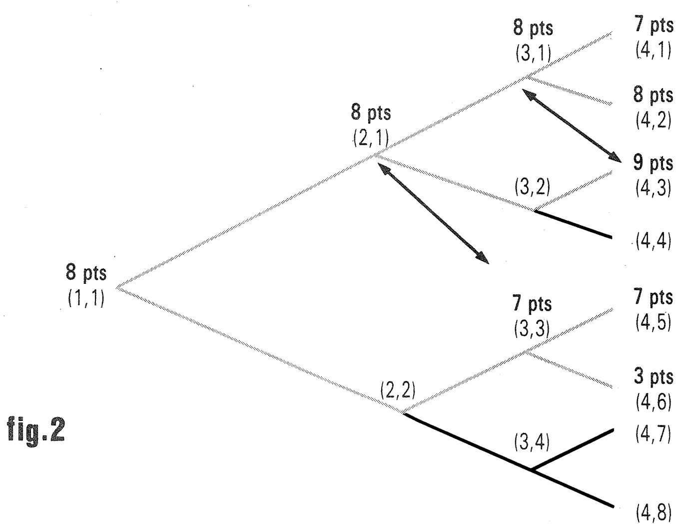
Sur l'exemple fig. 1, l'évaluation a noté chacun des coups du bout d'arbre : il est évident que la meilleure note est 12 points : pour se diriger vers ce coup, il faut que l'ordinateur joue (2,2). Cependant, rien ne permet d'affirmer que l'adversaire fera la bêtise de jouer (3,3) pour offrir ces 12 points à l'ordinateur ! Il est évident qu'il a intérêt à jouer (2,1), l'ordinateur ne laisse que deux coups possibles, respectivement à 8 et 9 points, tous deux bien meilleurs que les 4 points trouvés précédemment : on montre ainsi que, bien que le coup à 12 points ne se trouve dans la branche (2,2), le meilleur coup à jouer est bien (2,1) assurant à l'ordinateur au minimum 8 points.

Cet exemple met en évidence la méthode (nommée « mini-MAX ») pour remonter l'arbre. A tous les nœuds d'arbre impairs (là où l'ordinateur joue), il faut sélectionner la note MAXimale (car l'ordinateur choisit son coup), tandis qu'aux noeuds pairs (là ou l'adversaire joue), il faut sélectionner la note MINImale (car il faut supposer que l'adversaire choisira le coup qui s'avèrera le plus mauvais pour l'ordinateur !).

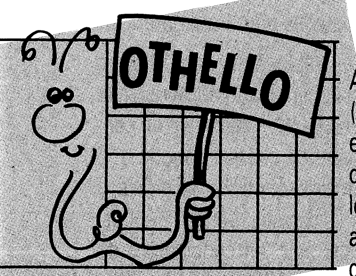
Mais on constate rapidement que l'arbre s'étend considérablement en un nombre limité de demi-coups de recherche. Ainsi, si l'on considère qu'en moyenne, chaque joueur a 6 coups, l'ordinateur évaluera 6^2 = 36 grilles, à 3 1/2 coups, 6^3 = 216 grilles... A 8 1/2 coups, il évaluera 6^8 grilles soit, en considérant un temps d'évaluation de 1/100 secondes, il faudrait environ 280 minutes soit 4 H 40 de réflexion par coup !

Pourtant, nos programmes jouaient en milieu de partie à 8 et neuf demi-coups de profondeur. En effet, à chaque recherche d'arbre, il est inutile d'analyser toutes les ramifications : Un petit exemple (fig 2) vous montre la principale méthode, nommée « Alpha-Bèta » d'élagage d'arbre.





Aux premières évaluations, les notes (4,1) et (4,2), ont été de 7 et 8 points : la maximale est donc de 8 points.



A l'évaluation (4,3), la meilleure est 9 points : il est donc évident que le programme peut arrêter l'évaluation de cette ramification qui ne ramènera aucune note que l'on pourra retenir !

En effet, après cette évaluation de 9 points, le noeud (3,2) sera affublé d'une note supérieure ou égale à 9, quoi qu'il arrive, la note (3,1) sera donc retenue face à la note du (3,2). Cette méthode pour élaguer s'avère dans la pratique très efficace (surtout quand les premiers coups essayés (4, 3) sont les meilleurs .

LA RECURSIVITE

Pour générer un arbre permettant à l'ordinateur de réfléchir à 8 1/2 coups, il serait évidemment ridicule d'imbriquer 8 routines de génération de coups : cette méthode serait fastidieuse et ne permettrait pas de paramètrer la profondeur de recherche.

Dans un tel cas, un programme récursif s'impose.

La routine de génération de tous les coups jouables d'une grille s'appelle donc elle même, jusqu'à ce que le nombre d'appels soit égal à un nombre fixé, en l'occurrence 8 dans notre exemple. Ceci permet, en ne changeant qu'un simple petit octet, de modifier la profondeur du jeu, et donc le temps de réponse !.

Dans notre programme, une petite variante a été adoptée : la génération des coups a été dédoublée pour des raisons de facilité, et surtout de rapidité : le programme de génération des coups blancs appelle le programme de génération des coups noirs, qui rappelle le premier etc... Ainsi en dédoublant pour les deux couleurs, le retour noeud, l'évaluation... nombre de tests ont été supprimés l'efficacité en a donc été d'autant plus accrue.

TRAITEMENT DU « PASSE »

Un problème se pose dans les programmes récursifs : le traitement du « passe ». Trois solutions sont possibles.

- La première consiste à penser que le « passe » est une très mauvaise chose à ce stade de l'arbre, on attribue donc à celui qui passe une très mauvaise note, et l'on remonte l'arbre : C'est une solution qui peut être acceptable en début de partie, où « passer » signifie le plus souvent être complètement bloqué.
- La seconde consiste à continuer la recherche après ce passe, mais en comptabilisant ce passe comme un 1/2 coup. C'est la solution adoptée en milieu de partie, où il est important pour l'évaluation de donner une note sur des grilles oû c'est toujours la même couleur qui a posé le dernier pion !
- La dernière consiste à continuer la recherche après ce passe, mais sans le compter comme un demi coup : c'est ce que nous faisons en fin de partie, où l'important est que tous les pions soient posés, quel que soit l'ordre !.

Dans ces deux derniers exemples, il est important de considérer le cas des deux passes consécutifs : il faut arrêter la recherche, et compter les pions, pour déterminer le vainqueur.

Une dernière prévision avant de passer à l'évaluation : tout programmeur d'OTHELLO en L.M, doit être persuadé de l'importance de l'optimisation de ses routines, et particulièrement de la routine de retournement.

Une petite étude nous a montré qu'un programme passait la majorité de son temps dans les retournements : cette partie du programme n'étant soumise à aucun critère objectif, il est donc important de travailler son optimisation.

EVALUATION

La principale qualité d'une évaluation appelée au bout d'un arbre de recherche à 8 1/2 coups doit être la concision : Mais tout le problème réside dans le fait que cette concision ne doit

pas impliquer médiocrité. La structure de base d'une évaluation BASIC, tourne autour de deux pôles fondamentaux.

• Prendre garde à la valeur « positionnelle » de chaque pion : c'est la méthode qui consiste, avec des tables à attribuer à chaque pion une note de position (note forte pour les coins, note faible pour les cases adjacentes aux coins...).

• Contrôler la prise des coins puis des bords (valeur « structurelle ») c' est la phase essentielle du jeu par excellence : acquérir un coin, c'est favoriser l'extension de sa couleur à partir de ce coin : une aide précieuse pour la victoire.

Un seul paramètre nous a permis de réunir d'une manière très efficace ces deux critères.

Envisageons un bord, un bord contient 6 cases, prenant trois positions différentes possibles : vide ; noir ou blanc un bord de 6 cases peut donc prendre 36 soit 72 positions différentes : la méthode consiste donc à former une table de 729 octets dans lesquels sont stockées les notes de tous les bords possibles.

L'accès au coin est permis par les bords, mais également par les diagonales. Nous avons donc constitué dans notre programme une deuxième table de 729 octets adaptée aux diagonales.

En cours de jeu, il suffit donc d'ajouter les 4 valeurs de bords et les deux valeurs de diagonales pour avoir une très bonne approximation du jeu actuel. On voit en effet la couleur des coins pris, on voit également si des coins vont être pris le prochain demi coup on contrôle l'extension sur les bords...

A notre second programme, nous avons rapporté d'autres paramètres pour affirmer l'évaluation.

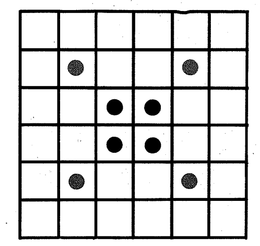
• Le premier consistait à contrôler la possibilité de jouer sur les cases B2, B5, E2, et E5.

En effet, ces cases adjacentes aux coins sont réputées dangereuses : mais environ au bout du 15e pion posé, le jeu commence à se bloquer, et l'un ou l'autre se voit contraint de jouer sur l'une de ces cases.

Le but de cette partie de l'évaluation est donc d'assurer à l'ordinateur une de ces cases diagonales pour bloquer le jeu, en ne laissant à l'adversaire que des cases diagonales offrant à l'ordinateur un coin sur un plateau.

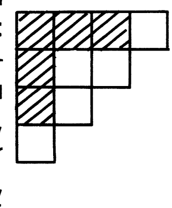
Le deuxième point important, allant de paire avec le premier, est le contrôle du centre. Lorsque les quatre pions du centre appartiennent à l'ordinateur, l'adversaire est le plus souvent bloqué. En effet jouer sur une case diagonale adjacente au coin implique immédiatement perdre le coin.

Quand le jeu se bloque, jouer une case diagonale (en vert), signifie donner un coin.



Le troisième paramètre essaie de combler un des défauts des grilles de 729, ces grilles aident la prise du coin, mais ne favorisent guère l'exploitation de ce coin.

Ainsi ce paramètre a pour but d'évaluer l'extension en triangle autour de ce coin : lorsque les 4 cases immédiatement adjacentes au coin, sont comme le coin, occupées par l'ordinateur (cases hachurées sur la fig.), cette routine évalue les 5 autres cases du triangle grâce à une table de valeurs (5 cases $3^5 = 243$ octets de grille).

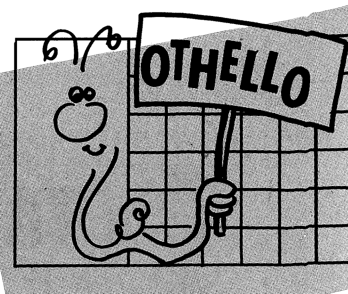


A ce stade des conseils, une restriction s'impose : tous ces renseignements permettent au programme de bien jouer, mais de bien jouer au milieu de partie uniquement.

La fin de partie ne pose aucun problème. En effet, lorsque le nombre de cases libres permet au programme d'envisager tous les coups jouables (environ 12 pour nos programmes), jusqu'à la fin du jeu, il suffit de remplacer l'évaluation précédemment décrite par un simple sous programme comptant les pions de chaque couleur.

Cette évaluation, parfaitement objective, permettra à l'ordinateur de finir la partie en jouant parfaitement tous les coups.

Le début de partie pose des problèmes plus sérieux. L'évaluation de milieu de partie s'avère relativement incompétente. En effet, au début, il est trop tôt pour que même à 8 demi-coups, quelque chose puisse se passer sur les coins...



Deux écoles s'affrontent : les uns sont partisans de la bibliothèque d'ouverture : cette méthode peut être efficace, mais également très dangereuse, il est en effet impératif que la bibliothèque soit adaptée au programme de milieu de partie, et cela, il est très difficile de bien le maitriser.

Les autres (et nous en faisons pour l'instant partie), préfèrent rédiger un programme adapté au début de partie, jouant intelligemment ce qui pourrait être enregistré en bibliothèque.

Jusqu'alors, aucune évaluation n'a réellement russi à nous satisfaire...

Voilà de tels conseils devraient vous permettre de gagner à des championnats d'Europe (BRUXELLES), ou aux championnats internationaux de Paris... (tel est notre palmarès).

Dès maintenant, à vos claviers : nous attendons avec impatience tous nos nouveaux adversaires lors de la prochaine rencontre d'OTHELLO.

A BIENTOT ?

Christophe CAMUS et Waël KOMBAR.

LES LOGICIELS « TANDEM/MICROVET »

UNE APPROCHE ORIGINALE DE L'INFORMATIQUE AGRICOLE

En agriculture, un des gros problèmes de l'installation de l'informatique dans les campagnes, reste la saisie des données sur le terrain. Si l'éleveur, le technicien d'élevage, l'inséminateur, le vétérinaire veulent obtenir de bons résultats, il faut qu'à tout moment ils puissent introduire dans leur machine des informations de qualité.

Les difficultés actuelles du monde de l'élevage ne permettent pas d'envisager des investissements importants. Heureusement les performances des PC SHARP ne cessent de s'améliorer et l'achat de ce matériel peu coûteux est bien souvent suffisant pour apporter en élevage des gains de productivité importants.

Le calcul optimum des ratio alimentaires permet de produire du lait ou de la viande au moindre coût. Le suivi des performances des animaux, de leur engraissement, de leur production laitière, de leur reproduction permet aux éleveurs une correction rapide, avant que la rentabilité de son exploitation ne soit altérée.

Les logiciels TANDEM permettent par l'utilisation de deux PC SHARP des échanges d'informations entre les éleveurs et leurs conseillers. Sur une SHARP PC 1253 préprogrammée l'éleveur saisit les informations issues de son élevage, et peut les valoriser très rapidement grâce à de petits logiciels simples à utiliser.

L'éleveur garde la maîtrise des informations qu'il collecte et peut s'il le désire, les faire traiter sur une SHARP plus puissante, (SHARP PC 1261), afin d'en obtenir un Bilan plus détaillé. Les deux machines (SHARP 1253 et SHARP 1261), peuvent être achetées par deux partenaires différents. En effet, informatique ne doit pas devenir synonyme de surabondance d'informations ; bien souvent l'éleveur demande des logiciels simples et pratiques et rechigne à investir dans un matériel qui lui fournira des informations qu'il est parfois incapable de commenter. Le technicien d'élevage, l'inséminateur, le vétérinaire sont prêts à conseiller les éleveurs mais reculent souvent devant le temps perdu à saisir les informations.

Les logiciels TANDEM sont une nouvelle forme de coopération entre l'éleveur et ses principaux conseillers. L'échange peut être fructueux pour les deux partenaires.

La première génération de logiciels TANDEM comprend 1 logiciel « carnet de reproduction vache laitière » associé à un logiciel « bilan de fécondité 365 jours ».

Le logiciel « carnet de reproduction » est un véritable agenda électronique qui enregistre au jour le jour les principaux événements dans l'exploitation. L'éleveur peut gérer ainsi un troupeau de 8 vaches et enregistre pour chaque vache une dizaine de paramètres (date de velage, date des inséminations, diagnostic de gestation...). A tout moment l'éleveur peut connaître les animaux à surveiller en diagnostic de gestation, velage etc... Cet agenda électronique lui permet donc de mieux veiller sur ses animaux, et de gagner du temps lors de leur remise à la reproduction. Chaque jour gagné représente de 10 à 15 F/jour de frais « d'hôtellerie » économisés. Cet agenda électronique libère le vétérinaire ou les conseillers de l'éleveur de la fastidieuse saisie des données, par le simple transfert du fichier de la SHARP 1253 dans une SHARP PC 1261 pourvu du logiciel approprié il est possible à tout moment, et rapidement (10 minutes maximum), d'effectuer un bilan de la reproduction dans l'exploitation sur les 365 derniers jours.

L'éleveur peut donc à tout moment être informé, d'une détérioration éventuelle des paramètres économiques de la gestion de la reproduction de son troupeau, et être rapidement conseillé par des professionnels compétents, quant aux mesures correctes à apporter. L'informatique de poche permet d'apporter des solutions performantes et bon marché, et même si les professionnels de l'informatique les méprisent un peu, il est évident que nous ne sommes qu'à l'aube d'une aarivée massive de ces petites machines dans le monde agricole.

TANDEM/MICROVET
21 rue Louis Melotte
92600 Asnières



```
BILAN/365J: LE.25.12.85.
****BILAN  FECONDITE****
DR *********************
------------------------
VACHES:    I VI1 VIF RET.
------------------------
VH: 111    2  53 113  39
VH: 222    4  32 104   9
VH: 333    2  32  73  20
VH: 444    2  53   0   0
VH: 555    2  32  74  21
VH: 666    1  73  73   0
VH: 777    3 104 250 104
VH: 211    1  60   0   0
************************
RESULTATS:   8 VACHES
************************
FECONDITE :
VELAGE-VELAGE MOY.= 384J
VEL-INS. FEC. MOY.= 114J
------------------------
VEL-I.F    RESULT. OBJEC.
  <40 J    0V   0%   0%
40-79 J    3V  50%  70%
80-110J    1V  16%  30%
 >110 J    2V  33%   0%
************************
FERTILITE :        MAXI.
I.C.(IA/IF)=  2.8 <1.6
TX.R.I1     1V  12%  70%
NB>=3IA     2V  25%  15%
************************
CONDUITE DU TROUPEAU :
************************
V-I1 MOYEN=  54J
------------------------
VEL-I.1    RESULT. OBJEC.
 <40 J    3V  37%   0%
40-69J    3V  37%  60%
70-90J    1V  12%  40%
 >90 J    1V  12%   0%
************************
ANALYSE DES RETARDS :
************************
I1-IF  MOYEN=  60J
RETARD MOYEN=  32J
TOTAL RETARD= 193J
------------------------
D.I.V.A 21 RUE L.MELOTTE
92600  ASNIERES (FRANCE)
```

ANNEXE 1

Date de la dernière utilisation de la PC 1253
Nom et l'adresse du vétérinaire traitant

CATALOGUE des ANIMAUX RECENSÉS

I :Nb d'inséminations
VI1 : Intervalle vélage-1ere ins.
VIF : Intervalle vélage-ins ; fec.
RET : Retard :
RET = (VIF-VI1)-(I-1)x21)

Coupez ici
BILAN

Intervalle vélage-vélage moyen
Intervalle vélage-insémination fécond.

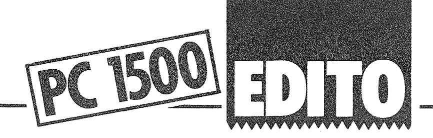
Ventilation des V-If.

Indice coïtal (Nb insémi./Nb Ins. Féc.)
Taux de réussite en première insémination
Nb de vaches ayant nécessité 3 ins. et +

Intervalle Vélage-1ere insémination moyen

Ventilation des V-I1

Intervalle 1ere ins.-ins. féc. moyen
retard moyen
Total des retards



Le PC-1211 à l'honneur. En effet ce trimestre nous vous présentons un long programme pour ce produit. Il s'agit d'un programme de navigation astronomique. Répondant à de nombreuses demandes, ce programme est transposable sur tous les PC. Désormais vous rencontrerez dans tous les bulletins des programmes BASIC facilement transposable d'un modèle à l'autre. Vous trouverez aussi des montagnes électroniques à faire sur vos pockets.
A bientôt. SHARPENTIEREMENT votre.

Luc BURELLIER



1500 MINITEL

Non, pas 1500 Minitels, mais un million pour la fin de l'année et même trois millions en 1986. Voilà une réalisation de prestige utile, accessible, qui ne devrait pas diviser le pays en deux clans... les minitellistes et des antiminitellistes. L'utilisation de votre PC 1500 favori comme centre serveur Minitel nécessite une légère modification de la CE-158. En outre il vous faudra confectionner un câble de liaison approprié. Bien entendu cette modification n'affecte en aucun point les caractéristiques de la CE-158, seulement deux nouvelles sorties seront créées sur le connecteur du RS 232C, et ainsi les broches 24 et 25 qui sont en l'air (c'est-à-dire non utilisées) seront activées.

Le Minitel sera relié au RS 232C par l'intermédiaire de sa prise péri-informatique (fig. 1).

Deux points nous amènent à modifer la CE-158 :

a) les signaux issus du Minitel sont inversés par rapport à ceux qu'accepterait la CE-158

b) le niveau de sortie (TX) 0, 6V est trop faible pour être reconnu par l'entrée (RD).

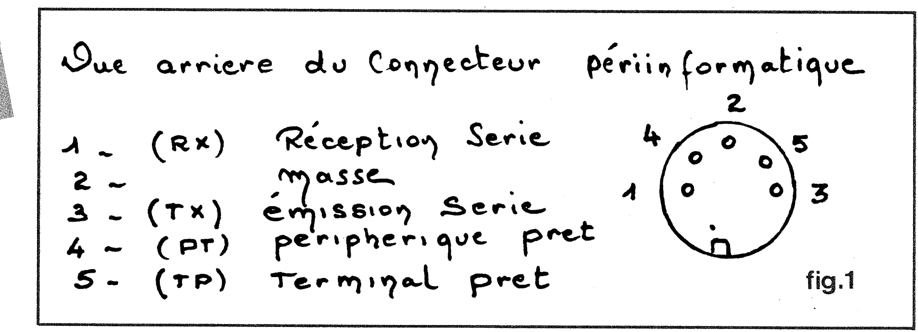


fig.1

Donc pour activer (RD) il faut amplifier (TX) et par bonheur la solution retenue nous inverse notre signal par la même occasion. Les signaux (TD) oscillent entre −6 et +6. C'est un bon niveau de sortie mais il faudra quand même les inverser (fig. 2). (CTS) sera activé par les 4V. venant de (PT). On ne s'occupera pas de (TP).

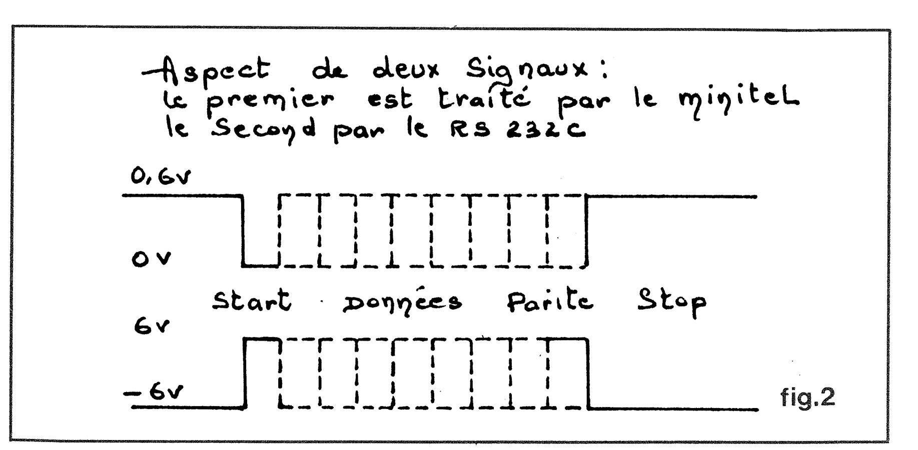


fig.2

Comme il est évident qu'un sharpentier ne souffrira jamais de se laisser envahir par des cartes et des modules, nous saisirons l'opportunité de glisser notre transformation au cœur de la CE-158. Ainsi BBX (car tel est son nom) sera alimenté par le VDD. Local qui supportera sans encombre les 10mA supplémentaires.

Ici attention ! Il ne faut pas confondre VDD et VC+. Notre VDD sera prélevé sur la patte 14 d'un des deux 74LS04 et la masse sur la patte 7 du même CI. Noter aussi que lors du démontage de la CE-158 la première intervention consiste à retirer les batteries. Tout CC risquerait de vous causer des désagréments.

N'ayant pas l'espace pour câbler notre circuit selon les techniques traditionnelles, nous ne compliquerons pas en gravant un CI et on soudera nos composants à plat, sur un morceau de vero-board de 40 × 20 mm. (les bandes dans le sens de la largeur) (voir fig. 3). A vous d'innover pour que BBX ne dépasse rn aucun cas 7 mm en hauteur.

Une fois câblé et ayant la mise en place il n'est pas interdit de la vérifier et même de le tester au multimètre.

Ensuite intervient la phase rigolote de la magouille qui consiste à coller BBX directement

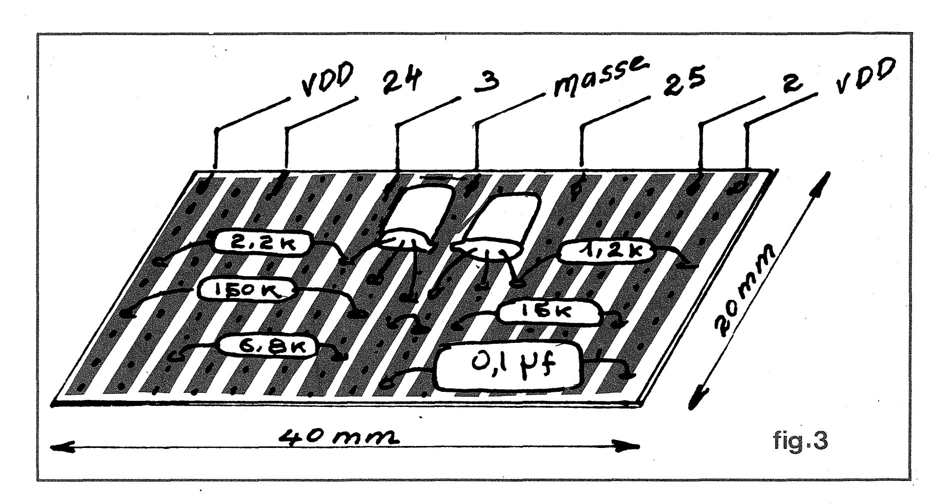


fig.3

sur l'UART de la CE, en l'occurence c'est un CDP 1854 ; pour ma part j'ai effectué la fixation avec un point de colle thermofusible, il me sera éventuellement ainsi possible de décoller ce circuit par une légère chauffe au fer à souder, il est inutile de préciser que de rôtir l'UART avec 12 bâtons de colle à chaud est préjudiciable au bon fonctionnement de la CE-158. Une goutte de la taille de la tête d'allumette sur le vero-board, on applique au bon endroit, c'est tout.

Sur les six fils issus de BBX on soude l'ALIM (masse et VVD). Passons à la prise cannon de 25(H) partie interne sur la broche numéro 2, on fait un piquage en y soudant le fil BBX2. Même chose en soudant BBX3 sur la broche 3. BBX24 broche 24, puis enfin BBX25 sur la broche 25.

Reste à remonter le tout sans oublier la batterie et on passe à la confection du câble, il sera prudent de ne pas dépasser 3M de câble blindé.

Inscription sur la touche	Code envoyé	Action avec la touche spéciale : codes et symboles ou séquence correspondante
ENVOI	1/3, 4/1	Retour chariot
RETOUR	1/3, 4/2	1/9, 4/2 (accent aigu)
REPETITION	1/3, 4/3	1/3, 4/A - Action T2 : cette séquence permet la procédure de correction d'erreur
GUIDE	1/3, 4/4	1/9, 4/8 (tréma)
ANNULATION	1/3, 4,5	5/C
SOMMAIRE	1/3, 4/6	1/9, 4/3 (accent circonflexe)
CORRECTION	1/3, 4/7	Action T1 : deux chiffres doivent préciser les vitesses d'échange avec un périphérique.
SUITE	1/3, 4/8	1/9, 4/1 (accent grave)
LOUPE	Action locale, pas d'émission en ligne	
A jusqu'à Z	4/1 jusqu'à 5/A	6/1 (a) jusqu'à 7/A (z) (lorsque le terminal est mode enseignement
ESPACE	2/0	7/F
CONTROLE	Aucune émission en ligne sans l'enfoncement simultané d'une autre touche	
CONNEXION-FIN	1/3, 4/9 séquence envoyée vers le modem du terminal exclusivement	1/3, 4/9 séquence envoyée vers la prise péri-informatique exclusivement

Inscription sur la touche	Code envoyé	Action avec la touche spéciale code et symbole correspondant
, (virgule)	2/C	3/C
. (point)	2/E	3/E
' (apostrophe)	2/7	4/0
; (point-virgule)	3/B	2/B +
–	2/D	3/D –
: (deux points)	3/A	2/A
? (point d'interrogation	3/F	2/F /
1	3/1	2/1 !
2	3/2	2/2
3	3/3	2/3
4	3/4	2/4 $
5	3/5	2/5 %
6	3/6	2/6 &
7	3/7	2/7
8	3/8	2/8 (
9	3/9	2/9)
0	3/0	5/E
*	2/A	5/B [
#	2/3	5/D]

MISE EN ROUTE

Embrocher le 1500 et la CE-158, puis vérifier que l'on peut encore l'utiliser comme avant. Sinon vous retournez à la case départ et ce qui suit ne vous concerne plus...

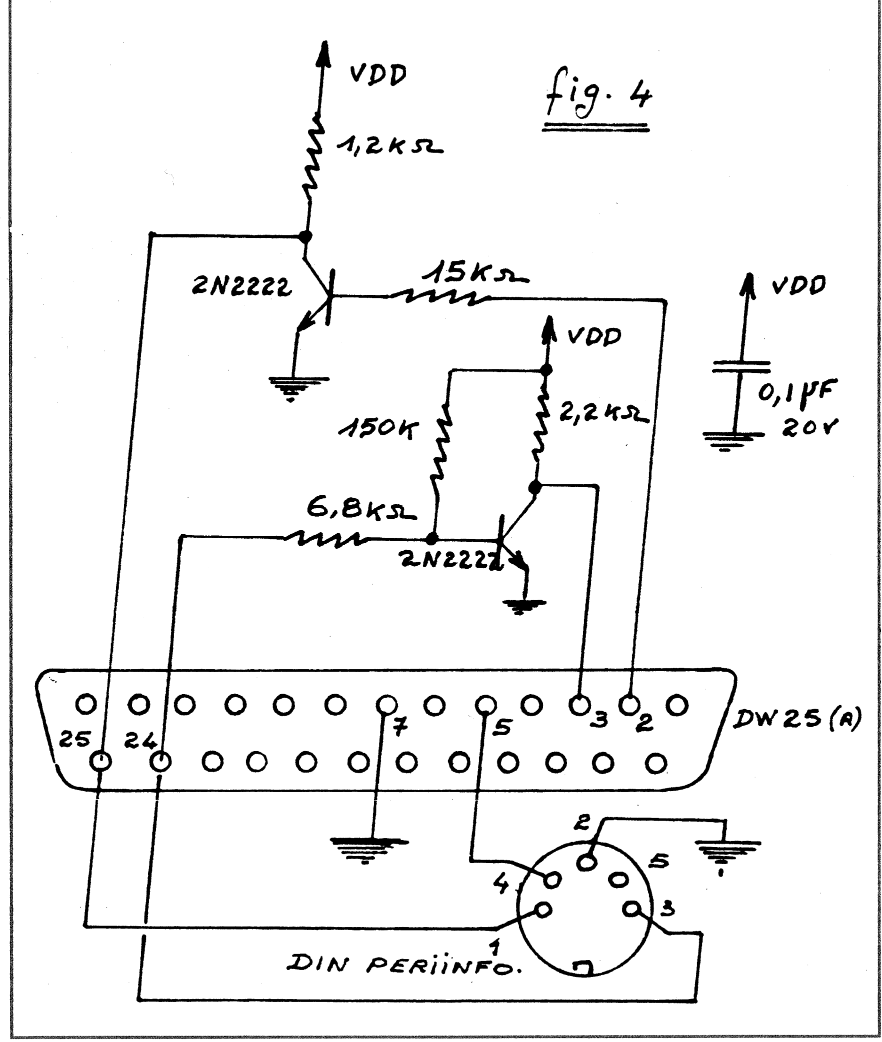
Stopper la CE et la raccorder par son nouveau câble au Minitel puis CE-158, ON, Sharp, ON. Tapez SETCOM 1200, 7, E, 1 ENTER terminal ENTER, mettre le Minitel sous tension, et constater que l'information passe effectivement dans les deux sens.

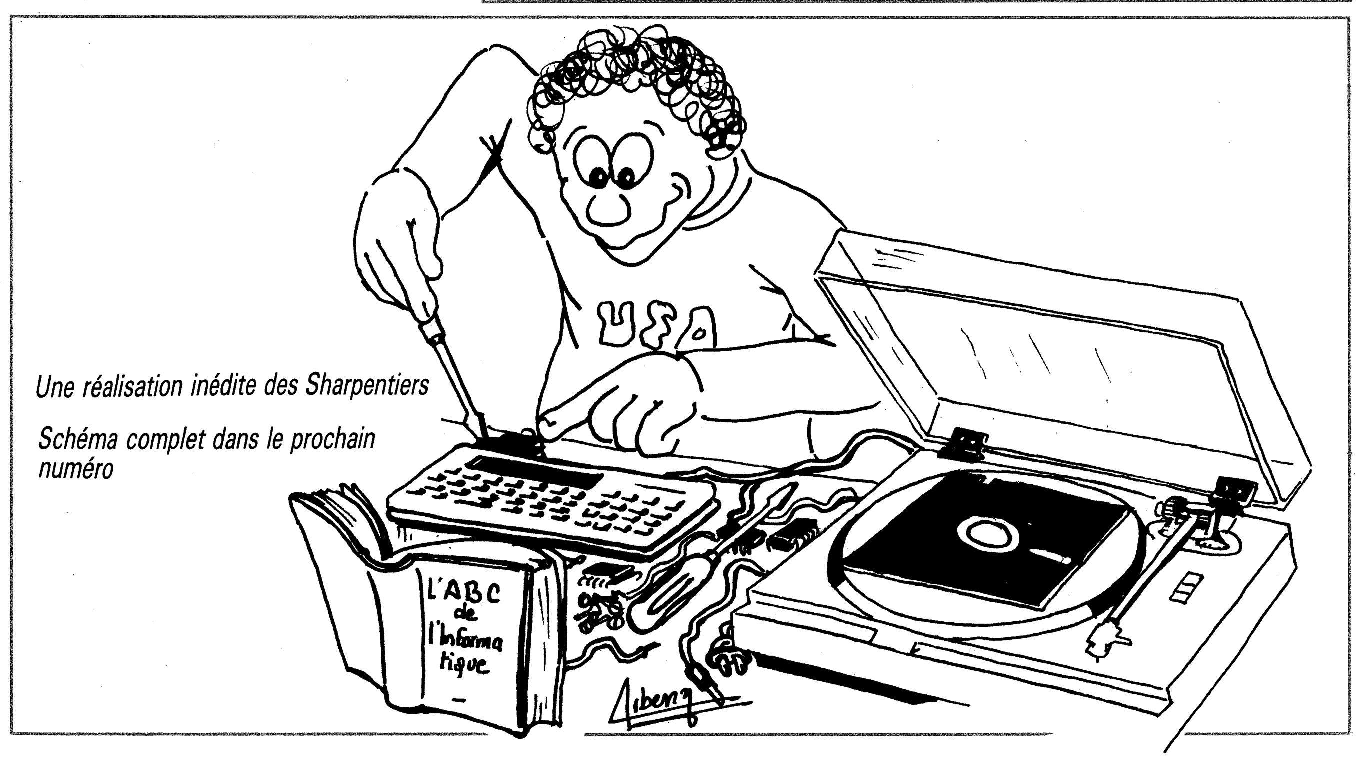
Sortir du mode terminal, SETDEV PO ENTER, histoire de faire un brin de toilette à l'écran. Console 80, 1, 0, envoyez LPRINT CHR$ &IB ; CHR$ &3A ; CHR$ &69 ; CHR$ &43 vous fera passer en mode ROULEAU. Vous pourrez ainsi lister au km. A quoi bon vous cacher les codes les plus croustillants, les séquences les plus ivres ! 1B ; 3A ; 69 ; 46 appelle la loupe. 1B ; 61 vous retourne la position du curseur.

Si j'envoie LPRINT CHR$ &1B ; CHR$ &48 ; I blink for Yoshino !
Avec 1B ; 61 si la porteuse est en ligne, le MODEM du Minitel se maintiendra ON, escape M, N, O, trois variantes des caractères dilatés CONTROL N, GRAPH, ETC.

Maintenant, dites-moi... ne pensez-vous pas que l'ETC pourrait faire l'objet d'une édition spéciale ?
A suivre...

MICHEL LOUTTER





Une réalisation inédite des Sharpentiers

Schéma complet dans le prochain numéro

LES CODES CACHES DU LH-5801

Dans toutes les tables publiées jusqu'à présent, on observe des colonnes entières d'instructions vides.

Par exemple :
la colonne &30-&3F ou celle &70-&7F.
Nous trouvons en effet devant un nouveau registre 16 bits que nous nommerons V, sa partie forte sera Vh et sa faible Vl.

Voyons tout d'abord comment remplir les cases.
Case &00 SBC Xl
Case &10 SBC Yl
Case &20 SBC Ul
La case 30 sera donc SBC Vl.

Autre exemple
la case &38, notée dans les cases actuelles NOP sera
Case &08 Xh
Case &18 STA Yh
Case &28 STA Uh
La case &38 sera donc STA Vh

Mais ces registres ont une particularité, c'est qu'ils sont insensibles à des affectations, la case &38 est donc bien un NOP car Vh ne prendra pas la valeur de A.

On s'aperçoit que Vl vaut toujours &00. Cela sera très utile :
LDA Vl &34 donne LDA 00
CPA Vl &36 donne CPA 00
ADI A,Vl &32 donne ADI A,c

Mais Vh ne suit heureusement pas cette destinée monotone, car lui évolue, il prend la valeur du poids fort de la dernière valeur 16 bits écrite et exécutée.

EXEMPLE
LDA (&n1 n2 n3 n4) STA (&n1 n2 n3 n4)
ORI (&N1 n2 n3 n4) BIA (&n1 n2 n3 n4)
Après l'exécution d'une de ces instructions, Vh vaudra &n1 n2.
L'utilisation des registres 16 bits ne modifie pas Vh.

Outre ces instructions, voici quelques cas particuliers où Vh est changé :
LDI S,&n1 n2 n3 n4 Vh=&n1 n2
JMP &n1 n2 n3 n4 correspond en fait à LDI P,&n1 n2 n3 n4donc Vh=&n1 n2
SPJ &n1 n2 n3 n4 correspond à PSH P
LDI P,&n1 n2 n3 n4 donc Vh=n1 n2
LDX V on aurait normalement Xh=Vh
Xl=V=&00 l.

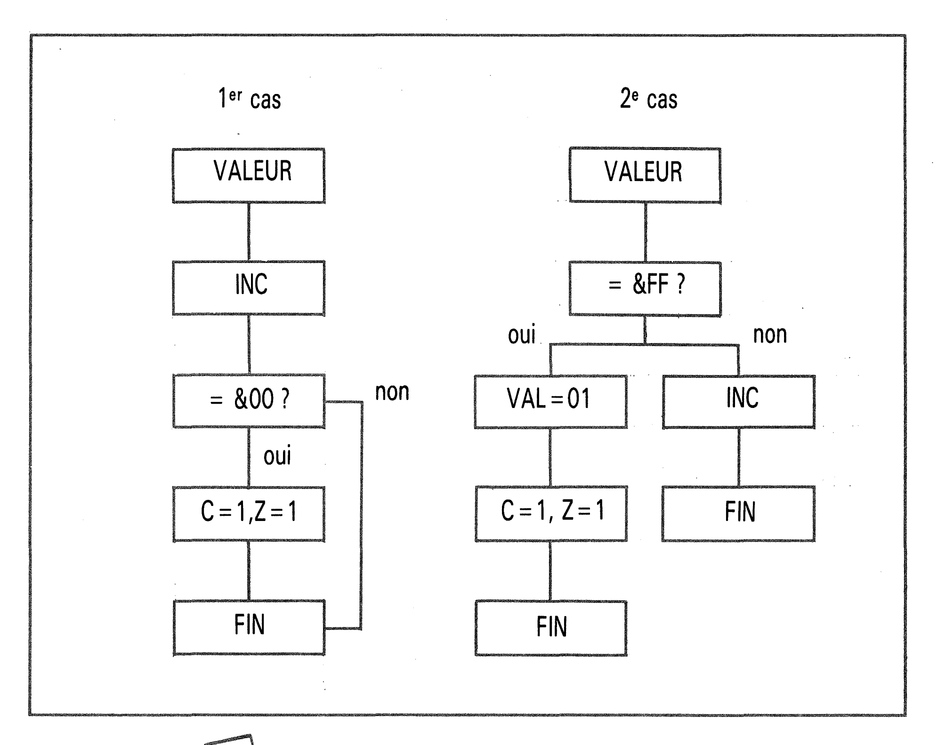
Or on s'aperçoit que Xh est inchangé, mais que XA vaut bien &00, cette instruction correspond donc à
LDI X1,&00

Voyons maintenant une particularité qui permet peut-être d'expliquer d'où provient ce registre V.
PSH V
On pourrait supposer que l'on va obtenir dans la pile, Vh puis V1, or on obtient dans cette pile :
(Poids fort de S – 1)(&00).
Et au retour V est inchangé, donc il s'est modifié au cours de l'exécution de cette instruction pour reprendre ensuite sa valeur originelle.

Il semble, de plus, que les instructions qui devraient mettre en valeur à l'adresse V ne fonctionne pas, par contre celles lisant un octet donnent le bon résultat.

Cette découverte va permettre de comprendre un peu mieux la marche exacte du microprocesseur, par exemple, lorsqu'il exécute une instruction INC, effectue-t-il l'opération puis positionne Z et C si le résultat est 0, ou bien teste-il- si le nombre à incrémenter est &FF, si oui il vaudrait 0 et il positionne C et Z, sinon il effectue l'opération. Et bien nous allons le savoir grâce à V. Voyons sous forme de schéma les deux cas.
Comment faire ? Rappelez-vous, Vh n'est pas modifié par les instructions INC et autres LD ou ST. On essayera donc INC VL, si c'est le premier cas, le INC n'a pas d'effet, donc Vl vaut toujours 0, il teste=0 ? oui –$ C=1 Z=1 or dans la réalité C=0 Z=0, vérifions le deuxième cas cette fois avec Vh valant &FF, on fera donc avant LDR (&FFFF) puis INC Vh, il teste =&FF ? oui –$ Vh=&00, mais il est insensible à ces instructions, donc Vh=&FF, C=1 Z=1, et c'est bien ce qu'on obtient.
On peut refaire ce test pour DEC ou d'autres instructions et cela permet de plonger au cœur microprocesseur.
Mais d'où vient V ? On peut supposer avec ce que l'on obtient avec PSH V que ce registre est un arctéfact d'un registre interne servant pour les calculs intermédiaires.

Marc Girondot



	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
0	SBC XI	SBC YL	SBC UI	SBC VI	INC XI	INC YI	INC UI	INC VI	SBC Xh	SBC Yh	SBC Uh	SBC Vh	VEJ C0	VEJ D0	VEJ E0	VEJ F0
1	SBX (X)	SBC (Y)	SBC (U)	SBC (V)	SIN X	SIN Y	SIN U	SIN V	BCR+	BCR–	SBC (ab)	SBI	VCR	ROR	SPU	AEX
2	ADC XI	ADC YI	ADC UI	ADC VI	DEC XI	DEC YI	DEC UI	DEC VI	ADC Xh	ADC Yh	ADC Uh	ADC Vh	VEJ C2	VEJ D2	VEJ E2	VEJ F2
3	ADC (X)	ADC (Y)	ADC (U)	ADC (V)	SDE X	SDE Y	SDE U	SDE V	BCS+	BCR–	ADC (ab)	ADI	VCS	DRR (X)	RPU	RPU
4	LDA XI	LDA YI	LDA UI	LDA VI	INC X	INC Y	INC U	INC V	LDA Xh	LDA Yh	LDA Uh	LDA Vh	VEJ C4	VEJ D4	VEJ E4	VEJ F4
5	LDA (X)	LDA (Y)	LDA (U)	LDA (V)	LIN X	LIN Y	LIN U	LIN V	BHR+	BHR–	LDA (ab)	LDI A	VHR	SHR	TIN	TIN
6	CPA XI	CPA YI	CPA UI	CPA VI	DEC X	DEC Y	DEC U	DEC V	CPA Xh	CPA Yh	CPA Uh	CPA Vh	VEJ C6	VEJ D6	VEJ E6	VEJ F6
7	CPA (X)	CPA (Y)	CPA (U)	CPA (V)	LDE X	LDE Y	LDE U	LDE V	BHS+	BHS–	CPA (ab)	CPI A	VHS	DRL (X)	CIN	CIN
8	STA Xh	STA Yh	STA Uh	STA Vh	LDI Xh	LDI Yh	LDI Uh	LDI Vh	LOP UI	LOPVI	SPV	RPV	VEJ C8	VEJ D8	VEJ E8	VEJ F8
9	AND (X)	AND (Y)	AND (U)	AND (V)	ANI (X)	ANI (Y)	ANI (U)	ANI (V)	BZR+	BZR–	AND (ab)	ANI A	VZR	SHL	ANI (ab)	REC
A	STA XI	STA YI	STA UI	STA VI	LDI XI	LDI YI	LDI UI	LDI VI	RTI	RTN	LDI S	JMP	VEJ CA	VEJ DA	VEJ EA	VEJ FA
B	ORA (X)	ORA (Y)	ORA (U)	ORA (V)	ORI (X)	ORI (Y)	ORI (U)	ORI (V)	BZS+	BZS–	ORA (ab)	ORI A	VZS	ROL	ORI (ab)	SEC
C	DSC (X)	DSC (Y)	DSC (U)	DSC (V)	CPI Xh	CPI Yh	CPI Uh	CPI Vh	DCA (X)	DCA (Y)	DCA (U)	DCA (V)	VEJ CC	VEJ DC	VEJ EC	VEJ FC
D	EOR (X)	EOR (Y)	EOR (U)	EOR (V)	BII (X)	BII (Y)	BII (U)	BII (V)	BVR+	BVR–	EOR (ab)	EAI	VHJ	INC A	BII (ab)	2nd
E	STA (X)	STA (Y)	STA (U)	STA (V)	CPI XI	CPI YI	CPI UI	CPI VI	BCH+	BCH–	STA (ab)	SPJ	VEJ CE	VEJ DE	VEJ EE	VEJ FE
F	BIT (X)	BIT (Y)	BIT (U)	BIT (V)	ADI (X)	ADI (Y)	ADI (U)	ADI (V)	BVS+	BVS–	BIT (ab)	BII A	VVS	DEC A	ADI (ab)	2nd

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
0	SBC XI	SBC YL	SBC UI	SBC VI	INC Xh	INC Yh	INC Uh	INC Vh	SBC Xh	SBC Yh	SBC Uh	SBC Vh	RDP	RDP	RDP	RDP
1	SBC (X)	SBC (Y)	SBC (U)	SBC (V)	SIN X	SIN Y	SIN U	SIN V	SIE	SIE	SBC (ab)	HTL	SDP	ROR	SPU	AEX
2	ADC XI	ADC YI	ADC UI	ADC VI	DEC Xh	DEC Yh	DEC Uh	DEC Vh	ADC Xh	ADC Yh	ADC Uh	ADC Vh	RDP	RDP	RDP	RDP
3	ADC (X)	ADC (Y)	ADC (U)	ADC (V)	SDE X	SDE Y	SDE U	SDE V	SIE	SIE	ADC (ab)	HLT	SDP	DRR (X)	RPU	RPU
4	LDA XI	LDA YI	LDA UI	LDA VI	INC X	INC Y	INC U	INC V	LDA Xh	LDA Yh	LDA Uh	LDA Vh	RDP	RDP	RDP	RDP
5	LDA (X)	LDA (Y)	LDA (U)	LDA (V)	LIN X	LIN Y	LIN U	LIN V	SIE	SIE	LDA (ab)	HTL A	SDP	SHR	TIN	TIN
6	CPA XI	CPA YI	CPA UI	CPA VI	DEC X	DEC Y	DEC U	DEC V	CPA Xh	CPA Yh	CPA Uh	CPA Vh	RDP	RDP	RDP	RDP
7	CPA (X)	CPA (Y)	CPA (U)	CPA (V)	LDE X	LDE Y	LDE U	LDE V	SIE	SIE	CPA (ab)	HTL	SDP	DRL (X)	CIN	CIN
8	LDX X	LDX Y	LDX U	LDX V	LDX S	LDX P	LDX S	LDX P	PSH X	PSH Y	PSH U	PSH V	PSH A	PSH A	PSH A	PSH A
9	AND (X)	AND (Y)	AND (U)	AND (V)	ANI (X)	ANI (Y)	ANI (U)	ANI (V)	SIE	SIE	AND (ab)	HLT	SDP	SHL	ANI (ab)	REC
A	POP X	POP Y	POP U	POP V	STX X	STX Y	STX U	STX V	POP A	POP A	TTA	ITA	ADR X	ADR Y	ADR U	ADR V
B	ORA (X)	ORA (Y)	ORA (U)	ORA (V)	ORI (X)	ORI (Y)	ORI (U)	ORI (V)	SIE	SIE	ORA (ab)	HLT	SDP	ROL	ORI (ab)	SEC
C	DSC (X)	DSC (Y)	DSC (U)	DSC (V)	OFF	OFF	OFF	OFF	DCA (X)	DCA (Y)	DCA (U)	DCA (V)	ATP	ATP	ATT	ATT
D	EOR (X)	EOR (Y)	EOR (U)	EOR (V)	BII (X)	BII (Y)	BII (U)	BII (V)	SIE	SIE	EOR (ab)	HLT	SDP	INC A	BII (ab)	2nd
E	STA (X)	STA (Y)	STA (U)	STA (V)	STX S	STX P	STX S	STX P	CDV	CDV	STA (ab)	RIE	AM0	AM1	AM0	AM1
F	BIT (X)	BIT (Y)	BIT (U)	BIT (V)	ADI (X)	ADI (Y)	ADI (U)	ADI (V)	SIE	SIE	BIT (ab)	HLT	SDP	DEC A	ADI (ab)	2nd

MODE D'EMPLOI DU PGM MONITEUR EN MNEMONIQUES OFFICIELS

Suite à de nombreuses demandes, nous publions un moniteur désassembleur en mnémoniques officiels. Il est entièrement en langage machine et occupe 4531 octets de mémoire vive. Il ne comprend pas les macro-instructions mais permet une recopie très facile sur imprimante. Il a été adapté d'un programme provenant de MICRO-SYSTEME.

MODE D'EMPLOI

Après avoir fait un CALL &4F00 vous passez en mode moniteur. Les fonctions sont :

M nnnn ENTER ENTER
Effectue un DUMP à l'affichage, les 4 curseurs sont utilisables et permettent de modifier des octets en mémoire.

L nnnn-mmmm
Effectue un DUMP sur imprimante par groupe de six octets.

A nnnn
Passe en mode assemblage, avec CA vous effacez le mnémonique si il y a erreur, et vous pouvez entrer le vôtre ; vous pouvez utiliser pour cela les touches réaffectées. Les touches curseurs sont bien entendu utilisables.

D nnnn
Désassemble le programme et permet aussi de modifier un mnémonique. Vous pouvez vous déplacer en mémoire avec les flèches. Le désassemblage vers le haut sera souvent faux, le programme remontant d'un octet quelque soient les octets précédents.

I nnnn-mmmm
Effectue un listing désassemblé sur l'imprimante

O nnnn+ (−) mmmm
Effectue une opération entre deux nombres héxadécimaux.

N nnnn mmmm oooo
Déplace les octets de nnnn à partir de oooo.

Q nnnn
Effectue un CALL &nnnn

De plus la touche « changement de réserve » copiera sur la CE-150 tous les caractères présents à l'affichage.

Pour avoir la signification exacte des touches du clavier, reportez-vous à la figure 1. Le listing du programme est donné de façon à être rentré avec le micro-moniteur SHARP du numéro 9. Si vous en voulez une version sur cassette, sachez qu'il se trouve dans la logithèque et qu'il « coûte » 1 point de bonus.

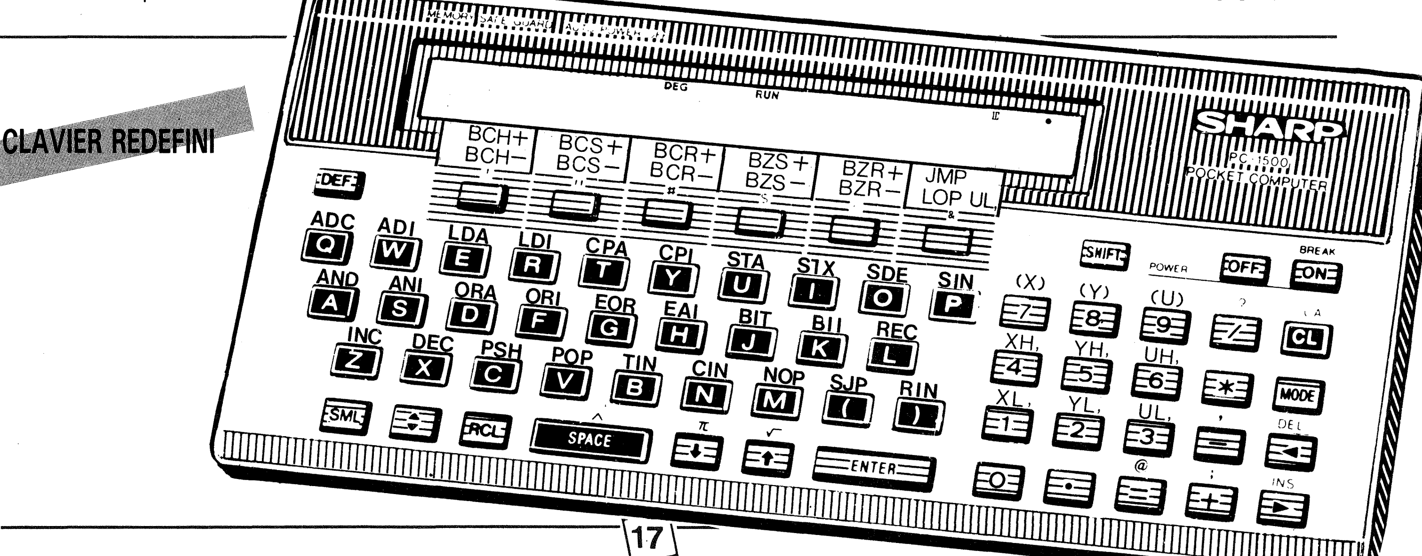
VICTOR OSTROMOURHON

```
 10 Input "DE", A, "A", Z
 20 H=0 I=INT (A/256) H=I GOSUB "HEX" LPRINT A$ ;
 30 I=AAND &FF H=H+I : GOSUB "HEX" LPRINT A$ ; ":";
 40 FOR A=ATO A+7 I=PEEK A :H=H+I : GOSUB "HEX" : LPRINT A$ ; " " ; : NEXT A
 45 I=HAND &FF : GOSUB "HEX" : LPRINT " " ; A$
 50 IF A(ZGOTO 20
 60 END
100 "HEX" J=INT (I/16) :K=IAND 15 : A$=CHR$ (J+48+7+*(J<9))+CHR$ (K+48+7*(K<9)) : RETURN
```

Ce petit utilitaire permet de sortir un listing de la mémoire sous une forme identique à celle du moniteur Sharp du bulletin numéro 9. Cela vous permettra de vérifier que vous n'avez pas fait d'erreur grâce à la somme de contrôle à la gauche des 8 cadres. Taper RUN et répondre aux deux bornes du dump.

MARC GIRONDOT

CLAVIER REDEFINI



```
4F00: BA 4F F7 48 5D 4A FB 38   71
4F08: 6A 04 68 04 BE 5A BC 3B   90
4F10: 22 FD C8 BE 56 DE FD 88   BD
4F18: 54 54 BE 56 8E FD C8 BE   34
4F20: 5E 18 FD 1A FD 2A 24 41   9B
4F28: 14 41 A4 41 94 41 85 04   3F
4F30: 8A 5E 25 85 3F BE 56 5E   22
4F38: 9A 14 06 81 02 94 86 9A   72
4F40: BE E6 69 5A 5E BE 56 DC   44
4F48: FD 88 BE 58 BA FD 2A A4   87
4F50: 96 83 0B 55 61 BE 4F 39   BF
4F58: 91 07 BE E6 69 9A 24 16   20
4F60: 91 0F FD 98 54 64 BE 4F   A9
4F68: 39 91 07 FD 0A 56 66 46   91
4F70: 38 57 63 94 86 99 06 14   7E
4F78: 06 99 0A BE E6 69 9A 1A   31
4F80: BE 58 BA FD 88 BE 58 03   40
4F88: BE 56 81 BE 57 D2 BE 57   69
4F90: C9 FD 0A BE 4F 39 91 15   99
4F98: 9A 68 55 6A 34 87 01 8B   1F
4FA0: 09 6A 48 87 1F 9B 03 BA   C8
4FA8: 57 9D FD A8 BE E6 69 BE   5B
4FB0: 57 16 BE 56 FC FD 2A 87   5A
4FB8: 01 89 04 BE E6 69 9A 87   F3
4FC0: 18 8B 18 1A 65 08 65 18   D1
4FC8: 64 65 87 00 9B 10 87 03   FC
4FD0: 9B 0E B7 04 9B 10 16 99   DD
4FD8: 11 25 0A BA 58 E7 85 0F   24
4FE0: AE 77 50 85 0E FD C8 85   E1
4FE8: 20 BE 56 5E FD 8A DF 99   C9
4FF0: 0C 85 0F AE 77 50 9A 85   D3
4FF8: 01 AE 79 F4 8A 57 86 38   32
5000: D8 52 4F 4C FB 53 45 43   EE
5008: D1 52 4F 52 E1 53 50 55   F5
5010: F1 41 45 58 E3 52 50 55   09
5018: D5 53 48 52 38 4E 4F 50   4F
5020: A8 53 50 56 88 52 50 56   C1
5028: D9 53 48 4C F9 52 45 43   0B
5030: 9A 52 54 4E F7 43 49 4E   DF
5038: CF 56 56 53 DF 44 45 43   81
5040: 41 DD 49 4E 43 41 44 49   56
5048: 4E 43 58 54 49 4E 43 59   03
5050: 64 49 4E 43 55 45 44 45   02
5058: 43 59 56 44 45 43 59 66   24
5060: 44 45 43 55 43 53 44 45   F0
5068: 58 53 53 44 45 59 63 53   4E
5070: 44 45 55 45 4C 49 4E 58   1E
5078: 55 4C 49 4E 59 65 4C 49   53
5080: 4E 55 47 4C 44 45 58 57   3E
5088: 4C 44 45 59 67 4C 44 45   42
5090: 55 40 49 4E 43 58 4C 50   43
5098: 49 4E 43 59 4C 60 49 4E   5E
50A0: 43 55 4C 42 44 45 43 58   3A
50A8: 48 52 44 45 43 59 48 62   61
50B0: 44 45 43 55 48 04 4C 44   FD
50B8: 41 58 4C 14 4C 44 41 59   2B
50C0: 4C 24 4C 44 41 55 4C 84   76
50C8: 4C 44 41 58 48 94 4C 44   AD
50D0: 41 59 48 A4 4C 44 41 55   CC
50D8: 48 06 43 50 41 58 4C 16   04
50E0: 43 50 41 59 4C 26 43 50   62
50E8: 41 55 4C 86 43 50 41 58   CC
50F0: 48 96 43 50 41 59 48 A6   39
50F8: 43 50 41 55 48 08 53 54   68
5100: 41 58 48 18 53 54 41 59   8B
5108: 48 28 53 54 41 55 48 0A   59
5110: 53 54 41 58 4C 1A 53 54   AE
5118: 41 59 4C 2A 53 54 41 55   B6
5120: 4C C0 56 45 4A 43 30 D0   A5
5128: 56 45 4A 44 30 E0 56 45   4D
5130: 4A 45 30 F0 56 45 4A 46   5B
5138: 30 C2 56 45 4A 43 32 D2   A7
5140: 56 45 4A 44 32 E2 56 45   69
5148: 4A 45 32 F2 56 45 4A 46   77
5150: 32 C4 56 45 4A 43 34 D4   C7
5158: 56 45 4A 44 34 E4 56 45   85
5160: 4A 45 34 F4 56 45 4A 46   93
5168: 34 C6 56 45 4A 43 36 D6   E7
5170: 56 45 4A 44 36 E6 56 45   A1
5178: 4A 45 36 F6 56 45 4A 46   AF
5180: 36 C8 56 45 4A 43 38 D8   07
5188: 56 45 4A 44 38 E8 56 45   BD
5190: 4A 45 38 CA 56 45 4A 43   9A
5198: 41 DA 56 45 4A 44 41 EA   58
51A0: 56 45 4A 45 41 CC 56 45   C3
51A8: 4A 43 43 DC 56 45 4A 44   CE
51B0: 43 EC 56 45 4A 45 43 CE   6B
51B8: 56 45 4A 43 45 DE 56 45   EF
51C0: 4A 44 45 EE 56 45 4A 45   FC
51C8: 45 00 53 42 43 58 4C 10   EA
51D0: 53 42 43 59 4C 20 53 42   53
51D8: 43 55 4C 80 53 42 43 58   BD
51E0: 48 90 53 42 43 59 48 A0   22
51E8: 53 42 43 55 48 02 41 44   35
51F0: 43 58 4C 12 41 44 43 59   5B
51F8: 4C 22 41 44 43 55 4C 82   A2
5200: 41 44 43 58 48 92 41 44   D1
5208: 43 59 48 A2 41 44 43 55   FD
5210: 48 D3 44 52 52 28 58 29   0E
5218: D7 44 52 4C 28 58 29 05   D1
5220: 4C 44 41 28 58 29 15 4C   4D
5228: 44 41 28 59 29 25 4C 44   5E
5230: 41 28 55 29 07 43 50 41   44
5238: 28 58 29 17 43 50 41 28   46
5240: 59 29 27 43 50 41 28 55   8C
5248: 29 08 4F 52 41 28 58 29   59
5250: 18 4F 52 41 28 59 29 28   74
5258: 4F 52 41 28 55 29 0E 53   93
5260: 54 41 28 58 29 1E 53 54   B5
5268: 41 28 59 29 2E 53 54 41   BB
5270: 28 55 29 01 53 42 43 28   69
5278: 58 29 11 53 42 43 28 59   B5
5280: 29 21 53 42 43 28 55 29   9A
5288: 03 41 44 43 28 58 29 13   61
5290: 41 44 43 28 59 29 23 41   88
5298: 44 43 28 55 29 09 41 4E   AF
52A0: 44 28 58 29 19 41 4E 44   CB
52A8: 28 59 29 29 41 4E 44 28   C8
52B0: 55 29 0D 45 4F 52 28 58   F3
52B8: 29 1D 45 4F 52 28 59 29   E0
52C0: 2D 45 4F 52 28 55 29 41   0C
52C8: 53 49 4E 58 51 53 49 4E   97
52D0: 59 61 53 49 4E 55 0C 44   6B
52D8: 43 53 28 58 29 1C 44 43   0C
52E0: 53 28 59 29 2C 44 43 53   35
52E8: 28 55 29 8C 44 43 41 28   5C
52F0: 58 29 9C 44 43 41 28 59   A8
52F8: 29 AC 44 43 41 28 55 29   8D
5300: 0F 42 49 54 28 58 29 1F   09
5308: 42 49 54 28 59 29 2F 42   55
5310: 49 54 28 55 29 F5 54 49   38
5318: 4E CD 56 4D 4A B5 4C 44   88
5320: 49 41 2C 87 43 50 49 41   FD
5328: 2C 88 4F 52 49 41 2C 48   01
5330: 4C 44 49 58 48 2C 58 4C   CC
5338: 44 49 59 48 2C 68 4C 44   DD
5340: 49 55 48 2C 4A 4C 44 49   C8
5348: 58 4C 2C 5A 4C 44 49 59   F7
5350: 4C 2C 6A 4C 44 49 55 4C   FF
5358: 2C 4C 43 50 49 58 48 2C   CB
5360: 5C 43 50 49 59 48 2C 6C   24
5368: 43 50 49 55 48 2C 4E 43   F1
5370: 50 49 58 4C 2C 5E 43 50   1D
5378: 49 59 4C 2C 6E 43 50 49   2F
5380: 55 4C 2C 8E 42 43 48 2B   26
5388: 9E 42 43 48 2D 83 42 43   78
5390: 53 2B 87 42 48 53 2B 8B   78
5398: 42 5A 53 2B 8F 42 56 53   7F
53A0: 2B 81 42 43 52 2B 8D 42   70
53A8: 56 52 2B 85 42 48 52 2B   5A
53B0: 89 42 5A 52 2B 93 42 43   8D
53B8: 53 2D 97 42 48 53 2D 9B   C7
53C0: 42 5A 53 2D 9F 42 56 53   89
53C8: 2D 91 42 43 52 2D 9D 42   8C
53D0: 56 52 2D 95 42 48 52 2D   96
53D8: 99 42 5A 52 2D 88 4C 4F   02
53E0: 50 55 4C 2C C3 56 43 53   FF
53E8: C7 56 48 53 CB 56 5A 53   C1
53F0: 8D 45 41 49 C1 56 43 52   7B
53F8: C5 56 48 52 C9 56 5A 52   C8
5400: B1 53 42 49 41 2C 83 41   44
5408: 44 49 41 2C B9 41 4E 49   E7
5410: 41 2C BF 42 49 49 41 2C   D1
5418: 48 4F 52 49 28 58 29 2C   76
5420: 58 4F 52 49 28 59 29 2C   9F
5428: 68 4F 52 49 28 55 29 2C   A3
5430: 4F 41 44 49 28 58 29 2C   76
5438: 5F 41 44 49 28 59 29 2C   8F
5440: 6F 41 44 49 28 55 29 2C   A3
5448: 49 41 4E 49 28 58 29 2C   92
5450: 59 41 4E 49 28 59 29 2C   AE
5458: 69 41 4E 49 28 55 29 2C   BF
5460: 4D 42 49 49 28 58 29 2C   AA
5468: 5D 42 49 49 28 59 29 2C   C3
5470: 6D 42 49 49 28 55 29 2C   D7
5478: 4C 4F 46 46 C0 52 44 50   99
5480: C1 53 44 50 8A 49 54 41   14
5488: CE 41 4D 30 DE 41 4D 31   05
5490: AA 54 54 41 EC 41 54 54   4C
5498: 81 53 49 45 81 49 4C 54   E7
54A0: CC 41 54 DE BE 52 49 45   D1
54A8: 8A 50 4F 50 41 9A 50 4F   5F
54B0: 50 58 1A 50 4F 50 59 2A   38
54B8: 50 4F 50 55 C8 50 53 48   03
54C0: 41 5E 53 54 58 50 88 50   DA
54C8: 53 48 58 98 50 53 48 59   EB
54D0: A8 50 53 48 55 18 4C 44   84
54D8: 58 58 28 4C 44 58 59 48   8D
54E0: 4C 44 58 53 58 4C 44 58   AF
54E8: 50 4E 53 54 58 53 5A 53   D9
54F0: 54 58 59 CA 41 44 52 58   42
54F8: DA 41 44 52 59 EA 41 44   C5
5500: 52 55 4A 53 54 58 58 6A   07
5508: 53 54 58 55 40 49 4E 43   C8
5510: 58 48 50 49 4E 43 59 48   D0
5518: 60 49 4E 43 55 48 42 44   CA
5520: 45 43 58 48 52 44 45 43   BB
5528: 59 48 62 44 45 43 55 48   E9
5530: FD 23 2F 2F 53 04 FF 11   6A
5538: 83 12 8D 13 A1 14 97 15   23
5540: 80 03 5D 03 FF 16 F7 03   87
5548: 53 04 FF 11 98 12 85 13   66
5550: C9 14 8F 15 D8 04 06 FF   37
5558: 16 DD 04 03 FF 37 03 38   18
5560: 0A 39 11 34 32 35 39 36   13
5568: 40 31 47 32 4E 33 55 03   80
5570: 5D 03 FF 51 CA 52 E8 41   8A
5578: CF 53 FB 47 D4 4A C0 48   5A
5580: E0 28 EE 44 E3 03 50 03   48
5588: FF 45 D3 55 FD 4F 64 4C   45
5590: 2C 5A 41 58 3C 42 4A 4E   1A
5598: 34 4D 1C 29 30 54 E5 03   1F
55A0: 52 03 FF 50 C7 03 53 03   89
55A8: FF 59 60 48 F0 03 54 03   47
55B0: FF 49 C1 43 BC 56 B2 57   6C
55B8: 38 46 20 00 00 00 00 00   AB
55C0: 00 00 00 00 00 00 00 00   15
55C8: 00 00 00 00 00 00 00 00   1D
55D0: 00 00 00 00 00 00 00 00   25
55D8: 00 00 00 00 00 00 00 00   2D
55E0: 00 00 00 00 00 00 00 00   35
55E8: 00 00 00 00 00 00 00 00   3D
55F0: 00 00 00 00 00 00 00 00   45
55F8: 00 00 00 00 00 00 00 00   4D
5600: 00 00 00 41 00 0C 18 24   DF
5608: 30 3C 48 06 12 1E 2A 36   A8
5610: 42 FD C8 48 74 A5 77 50   95
5618: B7 0E 81 05 48 76 F8 B1   23
5620: 0D B7 08 81 02 FD 40 58   5A
5628: 56 5A 03 2A 55 88 03 0A   45
5630: FD 8A 9A F2 49 77 4A 50   F2
5638: 85 01 41 85 20 6A 24 41   29
5640: 88 03 9A A5 77 50 2A 49   99
5648: 77 4A 51 44 88 03 9A 18   31
5650: BE 56 43 94 0E 9A BE 56   4D
5658: 43 05 9A BE 56 56 BE 56   0E
5660: 4F BE 56 11 CD 8A A5 77   9D
5668: 50 DD B7 18 89 04 BE 56   EE
5670: 69 9A AE 77 50 9A A5 77   F4
5678: 50 DF 9B 0E DF 9B 11 9E   CF
5680: 15 6A 24 58 77 5A 52 BE   92
5688: A4 F9 BE A9 F1 9A 55 BE   90
5690: 56 A7 0A B5 00 F9 83 10   5E
5698: 42 99 06 0A 55 BE 56 A7   E9
56A0: F9 02 0A AE 77 84 9A B7   F5
56A8: 40 81 03 FB 81 07 FB 81   21
56B0: 30 9A A5 77 90 4A 00 B7   6D
56B8: 10 81 06 40 FB B1 10 9E   3F
56C0: 0A BE 56 CF AE 77 82 04   AE
56C8: BE 56 CF AE 77 81 9A B7   F8
56D0: 0A 81 03 F9 83 07 F9 83   13
56D8: 30 9A 5A 55 58 77 55 B7   92
56E0: 20 9B 05 56 BE 56 BE 08   F6
56E8: AE 77 83 BE 56 BE 9A BE   E0
56F0: 56 DA BE 57 C0 9A 38 38   55
56F8: 38 38 38 38 BE E4 2C 93   8F
5700: 05 08 A5 77 51 86 89 09   E9
5708: BE 57 16 BE E4 2C 93 05   F0
5710: 08 84 AE 77 51 9A 5A 5F   8C
5718: 6A FF 89 02 52 99 07 9A   EE
5720: B7 09 89 04 BE 56 81 9A   F3
5728: B7 08 89 07 BE 56 58 BE   FB
5730: 56 76 9A B7 0C 89 04 BE   FB
5738: 56 5B 9A B7 2E 89 02 B5   FF
5740: 2C B7 18 9B 0B B7 3D 89   B5
5748: 02 B5 23 B7 60 93 15 B7   EF
5750: 11 89 02 B5 41 B7 12 89   88
5758: 02 B5 42 B7 13 89 02 B5   B2
5760: 43 B7 14 89 02 B5 44 B7   00
5768: 15 89 02 B5 45 B7 16 89   AF
5770: 02 B5 46 B7 20 91 3D BE   27
5778: 56 5E 9A 38 38 09 38 38   06
5780: 38 38 38 38 38 38 BE 56   3B
5788: 33 BE 59 50 BE 56 FC B7   40
5790: 18 9B 0D B7 0F 89 01 9A   91
5798: BE 57 9D 9E 14 B7 0D 89   A0
57A0: 0E BE 57 01 B7 0D 89 07   6F
57A8: BE 59 00 BE 56 33 9A BE   B5
57B0: 57 20 9A 56 5E B5 3A BE   79
57B8: 58 26 BE 57 D2 56 15 9A   79
57C0: A5 77 83 18 A5 77 84 1A   88
57C8: 9A 94 AE 77 83 14 AE 77   2E
57D0: 84 9A A5 77 85 19 95 77   18
57D8: B7 1A 9A 94 AE 77 86 14   8D
57E0: AE 77 87 9A B5 20 BE 56   66
57E8: 5E A5 77 81 BE 56 5E A5   51
57F0: 77 82 BE 56 5E 9A B5 56   60
57F8: 5E B5 3A BE 56 5E B5 20   E3
5800: BE 56 5E A5 77 83 BE 56   7D
5808: B5 BE 57 E9 A5 77 84 BE   71
5810: 56 B5 BE 57 E9 B5 20 BE   04
5818: 56 5E 9A BE 56 DA FD 88   31
5820: FD 1A BE 57 D8 9A BE 57   2E
5828: D2 54 BE 57 D8 56 15 BE   8F
5830: 56 B5 BE 57 E4 9A BE 56   3A
5838: EF FD 98 58 77 5A 5A 49   DF
5840: 07 55 87 20 98 05 FD 42   AA
5848: 88 0A BE 56 8F FD 2A 61   60
5850: FD A9 9E 13 FD 1A 9A BE   6D
5858: 58 18 BE 56 33 B5 40 BE   2A
5860: 57 F6 B5 06 58 77 5A 70   59
5868: FD 98 AE 77 85 BE 57 B7   CB
```

```
5870: FD 1A 51 A5 77 85 DF 99   49
5878: 11 9A BE 58 57 BE 56 FC   F8
5880: B7 0A 89 0B BE 57 D2 BE   D2
5888: 57 C9 BE 58 5A 9E 12 B7   D7
5890: 08 89 0D BE 57 D2 6A 0B   E3
5898: 56 88 03 BE 57 D8 9E 19   78
58A0: B7 0D 89 0E BE E6 69 BE   1E
58A8: 58 36 BE 56 FC B7 0D 99   FB
58B0: 31 9A BE 57 20 BE 59 50   6F
58B8: 9E 3D 5A 5A BE 56 DC FD   8C
58C0: 88 BE 58 18 FD 0A 9A BE   30
58C8: 58 8A FD 88 BE 57 D2 FD   98
58D0: 0A 94 96 81 05 14 06 81   6D
58D8: 01 9A BE 57 C9 FD 88 BE   EC
58E0: 58 5A BE 56 81 9E 18 44   7C
58E8: 45 FD 98 FD 98 BE 56 5E   11
58F0: FD 1A FD 0A FD 52 99 10   5E
58F8: 9A 00 00 00 00 00 00 00   EA
5900: A5 77 53 B7 4D 89 04 BE   17
5908: 38 7A 9A B7 4C 89 04 BE   1B
5910: 59 C7 9A B7 4E 89 04 BE   72
5918: 4F 40 9A B7 4F 89 04 BE   EB
5920: 5A 00 9A B7 44 89 04 BE   83
5928: 59 60 9A B7 41 89 04 BE   17
5930  59 85 9A B7 49 89 04 BE   6C
5938  4F 80 9A B7 51 89 03 BE   4C
5940: 4F 00 9A 00 00 00 00 85   37
5948: 0D 4A 09 48 04 8A 4D E6   3A
5950: A5 77 50 B7 1A 8B 07 BE   36
5958: 56 11 85 7F CD 8A 9A 38   75
5960: BE 5B 00 BE 57 16 BE 56   11
5968: FC B7 0A 89 0B BE 57 D2   F9
5970: BE 57 C9 BE 5B 03 9E 15   76
5978: B7 0B 89 09 BE 57 C0 56   50
5980: BE 57 DB 9E 15 B7 0D 98   CB
5988: 0A BE 4F 99 BE 59 50 9E   94
5990: 2B BE 57 C0 BE 57 D8 BE   97
5998: 5E 34 BE 57 16 BE 56 FC   BE
59A0: B7 0D 99 38 9A BE 58 1B   5C
59A8: BE 59 F2 BE 5A A8 BE 57   E2
59B0: 16 BE 56 FC B7 0A 89 13   8C
59B8: BE 57 D2 A5 77 61 B7 20   4C
59C0: 89 04 54 9E 57 DB BE 57   FF
59C8: C9 9E 23 B7 0B 89 06 BE   8A
59D0: 57 C0 56 9E 12 B7 0D 8B   95
59D8: 08 BE 4F 99 BE 59 50 9E   E4
59E0: 30 BE E6 69 BE 5E 34 BE   84
59E8: 57 16 BE 56 FC B7 0D 99   1B
59F0: 3D 9A BE E6 69 BE 56 33   74
59F8: BE 58 03 9A 80 80 80 80   04
5A00: BE 59 8A A5 77 59 B7 2B   81
5A08: 89 1C BE 5A 30 BE 57 C9   2D
5A10: 85 3D BE 57 F8 BE 56 FC   7C
5A18: B7 0D 89 01 9A B7 09 99   83
5A20: 0C BE 56 81 9E 11 B7 2D   AE
5A28: 8B 01 9A 9E 5A 38 9E 23   3C
5A30: 54 46 4C 00 99 06 4E 00   5D
5A38: 99 0A 9A 68 00 6A 00 54   05
5A40: 56 14 06 99 06 94 86 99   5C
5A48: 0A FD A8 FD 1A 9A 28 55   7F
5A50: B7 20 9B 05 56 9A BE 57   26
5A58: D2 54 BE 57 DB 56 9A 38   F0
5A60: 38 38 38 07 9B 0C FD A8   A5
5A68: 44 FD 62 99 05 FD 2A 88   92
5A70: 0E 9A BE 58 E7 14 18 85   50
5A78: 20 BE 58 E9 85 FF 9A 00   3F
5A80: 45 FD C8 FD A8 58 77 5A   82
5A88: 61 55 B7 20 9B 05 07 89   9F
5A90: 0A 44 FD 62 99 0D FD 2A   64
5A98: FD 8A 9A 44 FD 62 99 05   54
5AA0: FD 2A FD 8A 62 99 27 85   7F
5AA8: FF 9A 38 85 0E AE 77 50   0B
5AB0: 85 3A BE 56 5E 9A BE 5A   1D
5AB8: 63 B7 FF 9A BE 5A 80 B7   14
5AC0: FF 9A 85 08 AE 77 50 BE   A3
5AC8: 57 D2 54 BE 57 DB 56 15   FA
5AD0: BE 56 85 BE 57 E9 9A BE   49
5AD8: 5A C2 BE 5A A8 A5 77 80   AD
5AE0: 9A A5 77 50 DF AE 77 50   94
5AE8: 9A 09 BE 56 8E FD C8 84   CF
5AF0: FD C8 5A 53 BE 56 DC FD   A9
5AF8: 8A 41 FD 8A 8A 5A A9 0D   6E
5B00: BE 59 18 55 AE 77 80 BE   44
5B08: 57 DB BE 59 F2 BE 56 82   64
5B10: BE 57 E9 BE 5A AB A5 77   48
5B18: 80 B7 FD 89 03 BA 5D 48   92
5B20: 48 50 4A 00 6A 0E 68 04   41
5B28: 58 03 5A 01 BE 5A 86 89   90
5B30: 01 9A 5A 10 68 05 50 BE   1B
5B38: 5A B6 9B 08 6A 3F 68 06   60
5B40  58 03 50 BE 5A B6 9B 0E   BD
5B48  6A 19 68 07 5A 04 BE 5A   0B
5B50: B6 9B 0D 6A 02 68 05 5A   3C
5B58: 02 BE 5A B6 9B 0D 48 52   C5
5B60: 4A D6 6A 08 68 07 58 03   17
5B68: 5A 04 BE 5A B6 9B 13 6A   07
5B70: 00 68 04 58 03 5A 01 BE   98
5B78: 5A B6 89 01 9A BE 5A D7   F6
5B80: 9E 0A BE 5A E1 BE 57 D2   53
5B88: BE 5A CE 9A 48 53 4A 19   61
5B90: 6A 00 68 04 58 03 5A 01   77
5B98: BE 5A B6 9B 19 6A 02 68   49
5BA0: 06 58 03 5A 03 BE 5A B6   87
5BA8: 9B 0F 6A 0B 68 07 5A 04   EF
5BB0: BE 5A B6 9B 0D 58 04 6A   47
```

```
5BB8: 00 68 05 5A 01 BE 5A B6   A9
5BC0: 89 1C BE 57 C0 FD 98 BE   E8
5BC8: 57 D2 56 55 48 00 0A BE   07
5BD0: 5A 30 BE 57 C9 BE 58 03   AC
5BD8: FD 1A BE 57 C9 9A 6A 00   2C
5BE0: BE 5A B6 89 10 BE 57 C0   77
5BE8: FD 98 BE 57 D2 56 55 56   C0
5BF0: DF 99 04 9E 23 6A 07 68   61
5BF8: 05 58 04 BE 5A B6 9B 3E   5B
5C00: 6A 07 BE 5A B6 9B 22 38   90
5C08: 38 38 38 38 58 03 68 07   0E
5C10: 5A 04 6A 00 BE 5A B6 9B   9D
5C18: 34 38 38 38 38 38 38 38   30
5C20: 38 38 9B A2 5A 06 68 04   05
5C28: 5A 02 58 03 BE 5A B6 9B   A4
5C30: 0F 6A 03 68 06 5A 04 BE   92
5C38: 5A B6 9B 1A 6A 08 68 08   3E
5C40: 5A 06 BE 5A B6 9B 25 38   C2
5C48: 38 38 38 38 38 38 38 38   54
5C50: B5 0A BE 5A C4 BE 5A AB   0A
5C58: A5 77 80 48 5D 4A C0 5A   69
5C60: 09 68 05 58 03 5A 03 BE   9F
5C68: 5A B6 89 13 BE 5A E1 BE   27
5C70: 57 D2 56 BE 5A CE BE 5B   4A
5C78: 85 85 29 BE 56 5E 9A 6A   9D
5C80: 03 68 05 58 03 BE 5A B6   75
5C88: 9B 1E 6A 02 BE 5A B6 9B   72
5C90: 25 6A 00 68 06 5A 04 BE   05
5C98: 5A B6 89 0C BE 5C 6C BE   DD
5CA0: 5A E1 85 20 BE 56 5E 9A   18
5CA8: 6A 00 68 04 5A 01 BE 5A   4D
5CB0: B6 89 05 BE 5C 6F 9E 19   90
5CB8: 6A 00 68 05 5A 03 BE 5A   50
5CC0: B6 89 0E BE 5C 6C 38 38   5F
5CC8: 38 38 38 38 38 38 38 38   E4
5CD0: 9A 6A 00 68 04 5A 01 BE   85
5CD8: 5A B6 9B 29 85 0C BE 5A   E1
5CE0: C4 BE 5A A8 A5 77 80 48   A7
5CE8: 5D 4A F8 6A 02 68 05 58   17
5CF0: 04 5A 01 BE 5A B6 89 20   22
5CF8: BE 5A E1 BE 57 D2 56 56   E0
5D00: BE 5A CE BE 57 D2 56 BE   3E
5D08: 5A CE 85 29 BE 56 5E 85   92
5D10: 2C BE 56 5E BE 56 85 9A   43
5D18: 6A 00 68 04 58 04 BE 5A   8F
5D20: B6 9B 2B 85 08 AE 77 50   2B
5D28: B5 06 FD C8 85 20 BE 56   EE
5D30: 5E FD 8A DF 99 0C BE 57   0B
5D38: D2 56 56 56 BE 57 DB BE   17
5D40: 5A A8 85 3F BE 56 5E 9A   A2
5D48: BE 5A D7 BE 57 D2 56 15   E6
5D50: AE 77 80 48 54 4A 78 6A   1A
5D58: 0B 68 04 58 03 5A 01 BE   A0
5D60: 5A B6 89 01 9A 6A 13 68   D6
5D68: 05 58 03 5A 02 BE 5A B6   4F
5D70: 9B 0E 6A 05 68 06 58 03   AE
5D78: 5A 03 BE 5A B6 9B 18 BE   74
5D80: 57 D2 56 38 BE 57 DB 85   39
5D88: FD AE 77 80 48 55 4A 30   9E
5D90: 5A 00 68 02 58 01 5A 01   75
5D98: BE 5A B6 BE 5A A8 9A 38   59
5DA0: 38 38 38 38 38 38 38 38   8D
5DA8: 38 38 38 38 38 38 38 38   C5
5DB0: 38 38 38 38 38 38 38 38   CD
5DB8: 38 38 38 38 38 38 38 38   D5
5DC0: AF 42 49 54 28 A1 53 42   09
5DC8: 43 28 A3 41 44 43 28 A9   CC
5DD0: 41 4E 44 28 AD 45 4F 52   8B
5DD8: 28 A5 4C 44 41 28 A7 43   E5
5DE0: 50 41 28 AB 4F 52 41 28   AB
5DE8: AA 4C 44 49 53 2C BE 53   58
5DF0: 4A 50 AE 53 54 41 28 BA   5F
5DF8: 4A 4D 50 E9 41 4E 49 28   25
5E00: ED 42 49 49 28 EF 41 44   9B
5E08: 49 28 EB 4F 52 49 28 2F   03
5E10: 2F 2F 2F 2F 2F 2F 2F 2F   E6
5E18: 2F 2F 2F FD C8 5A 53 BE   33
5E20: 56 DC FD 8A 9A FD C8 BE   54
5E28: 58 1E FD 89 56 DF 99 04   55
5E30: BE 5B 03 9A 48 54 4A 78   A2
5E38: 5A 0C 58 03 BE 5A 8C 9B   D6
5E40: 0D BE 5E 18 18 85 FD 41   ED
5E48: 94 41 B5 02 9E 29 38 38   69
5E50: 38 38 38 38 38 38 38 38   6E
5E58: 38 48 55 4A 0C 6A 06 69   89
5E60: 05 BE 5A 8C 99 25 4A 8C   5B
5E68: 38 6A 02 68 07 BE 5A 8C   AD
5E70: 99 31 8B 08 BE 5E 18 41   A3
5E78: B5 01 9E 30 48 50 4A 00   3C
5E80: 5A 0F 68 03 BE 5A 8C 99   2F
5E88: 15 4A 91 6A 40 68 05 BE   AB
5E90: 5A 8C 99 0D 38 48 53 4A   C7
5E98: 15 6A 01 68 03 BE 5A 8C   85
5EA0: 99 10 48 50 4A 3C 6A 11   40
5EA8: 68 04 BE 5A 8C 99 1D 48   44
5EB0: 54 4A A8 6A 14 68 04 BE   FC
5EB8: 5A 8C 88 03 2A 5E 41 49   5B
5EC0: 52 4A 11 6A 19 68 06 BE   78
5EC8: 5A 8C 99 2C 6A 03 68 04   DA
5ED0: BE 5A 8C 99 08 6A 09 69   81
5ED8: 06 BE 5A 8C 99 14 48 5D   62
5EE0: 4A C0 6A 08 68 04 BE 5A   3E
5EE8: 8C 8B 21 FD C8 BE 56 DE   65
5EF0: FD 88 BE 5E 18 FD 1A FD   1E
5EF8: 8A 41 94 41 14 41 85 03   03
```

```
5F00: 8A 5E 25 38 38 38 38 38   84
5F08: 38 38 38 38 6A 01 68 05   1F
5F10: BE 5A 8C 99 2A 6A 01 68   D9
5F18: 03 BE 5A 8C 99 33 6A 01   85
5F20: 69 04 BE 5A 8C 99 3C 6A   F5
5F28: 01 68 03 BE 5A 8C 99 45   A5
5F30: 48 53 4A 83 6A 12 68 04   DF
5F38: BE 5A 8C 88 38 AE 77 80   D6
5F40: BE 56 DC FD 88 5A 53 BE   7F
5F48: 56 DC FD 1A 94 86 83 1E   A8
5F50: A5 77 80 F9 38 38 AE 77   D9
5F58: 80 FD 98 FD 88 FD 1A FD   65
5F60: 0A 54 54 BE 5A 3B A5 77   E0
5F68: 80 28 14 BA 5F 96 89 04   BF
5F70: 04 16 93 24 56 56 9E 15   FF
5F78: 4A DD 6A 01 68 06 BE 5A   EF
5F80: 8C 99 46 48 53 4A E4 5A   AD
5F88: 07 68 03 BE 5A 8C 9B 1F   D7
5F90: BE 5A 4E BE 56 BE FD C8   BC
5F98: A4 6C 98 89 02 85 88 FD   64
5FA0: C8 BE 5E 18 FD 8A 41 FD   C3
5FA8: 8A 41 85 02 BA 5E 25 6A   30
5FB0: 04 69 05 BE 5A 8C 99 28   15
5FB8: 6A 0C 68 07 BE 5A 8C 99   69
5FC0: 31 48 53 4A 19 5A 01 68   21
5FC8: 03 BE 5A 8C 99 3E 6A 03   42
5FD0: 68 05 BE 5A 8C 99 47 6A   8A
5FD8: 0C 69 06 BE 5A 8C 99 50   6E
5FE0: 48 55 4A 30 6A 01 68 01   2A
5FE8: BE 5A 8C 8B 03 BA 5E 74   35
5FF0: 8A 4F 03 38 38 38 38 38   73
5FF8: BE 57 D2 56 BE 57 DB 9A   1E
```

INVERSION VIDEO PARTIELLE ET APPLICATIONS

L'écran du PC-1500 n'est pas particulièrement grand, il convient donc de l'utiliser au maximum de ses possibilités et même au-delà (!), c'est ce que nous proposons de faire.

On trouve dans le TECHNICAL REFERENCE MANUAL un exemple d'inversion vidéo, mais elle fait uniquement une inversion totale sans pouvoir définir un nombre de colonnes ni leur place. C'est ce que nous vous proposons, avec ensuite une application qui reprend un peu le principe de la souris déplaçant un symbole sur des cases. La souris n'ayant pas été encore adaptée au PC-1500, nous nous contenterons du déplacement d'un symbole. Il sera symbolisé par une zone de l'écran en inversion que l'on pourra déplacer sur plusieurs choix et taper ENTER quand on aura la bonne option.

Le programme en langage machine est relogeable, c'est-à-dire que l'on peut le mettre à n'importe quel endroit en mémoire sans avoir à changer les codes en mémoire. Seul bien sûr l'argument du CALL est à modifier en conséquence.

La liste des codes est donnée dans le listing I, les DATA sont interprétés par le programme "POKE.LM" donné dans le listing 2. Pour utiliser ce dernier il faut d'abord opérer un RESTORE au numéro de la ligne où se trouve le premier DATA à traduire, et mettre dans Q l'adresse du premier octet où devra être implantée la routine. Pour la lancer on fera GOSUB "POKE.LM" qui pokera jusqu'à DATA "END".

Le programme est alors entré et utilisable.

Il faut alors définir les colonnes à inverser. On fera POKE &774E, n, m. n sera le numéro de la première colonne et m le numéro de la dernière, les colonnes étant numérotées de 0 à 155 et faire alors un CALL au premier octet du programme.

Passons maintenant aux applications, le choix par cochage. Dans tous les cas on utilisera la sous-routine de cochage qui est un utilitaire permettant d'utiliser au mieux ce principe. Il faut mettre dans A$, B$, C$ etc. les messages à afficher, la dernière variable étant vide. Pour indiquer la fin des possibilités, mettre dans W le premier octet de la routine et lancer la routine par GOSUB "COCHEZ" le programme est donné sur le listing 3. Pour l'utilisation, il faudra déplacer la zone inversée avec les flèches du curseur et appuyer sur ENTER quand la zone inversée est sur l'option choisie. Au retour du sous-programme vous aurez dans J le numéro correspondant à la variable qui a été choisie.

Par DEF X vous essayerez un programme de détermination d'une date et par DEF Z un programme de codage des cartes au bridge. Les programmes sont donnés sur le listing 4.

Cette méthode, outre un intérêt esthétique certain permet d'économiser des places de caractères car il n'est plus nécessaire de laisser des espaces entre les mots.

PAUL HENRY

LISTING 1

```
 90 "L" RESTORE 140:Q=&7150:GOSUB "POKE.LM":END
100 REM INV-VIDEO Partielle
110 REM
120 REM PREPA : No first Point ——► &774E No last Point ——► &774F (0-155)
130 REM
140 DATA "A5774E.0A      "REM LDA &774E:STA XL
150 DATA "A5774F.AE7875 "REM LDA &774F:STA &7875
160 DATA "FB.00.AE7800   "REM SEC :SBC X1 :STA &7800 Nb Points ——► &7800
170 DATA "CD8C.44        "REM MACRO &8C :INC X   last Point ——► Xr
180 DATA "5878.5A01      "REM  LDIYh &78: LDIYl &01 &7801 ——► Yr
190 DATA "B5F0.1E        "REM LDIA &F0: STA(Y) &F0 dans &7801
200 DATA "84.B902.8903   "REM LDAXh: ANI&02: BZR+3 si Xh=77ou76 on saute
210 DATA "B50F.1E        "REM LDIA &0F: STA(Y) sinon "0F" dans &7801
220 DATA "A57800.2A      "REM LDA &7800: STA Ul BOUCLE sur nb Points
230 DATA "05.1D.43       "REM LDA(X): EOR(Y): SDE(X) inverse digit bas
240 DATA "05.1D.43       "REM LDA(X): EOR(Y): SDE(X) inverse digit haut
250 DATA "4EFF.890A.4A4D "REM CPIXL FF: BZR+10: LDIXL 4D si chgt afficheur——►
260 DATA "84.B902.8903   "REM &4D  ►XL, test si Xh=&74ou&75==) &0F  ►&7801
270 DATA "B50F.1E        "REM LDIA &0F: STA(Y)
280 DATA "8816.9A        "REM test fin de boucle, RTN
290 DATA "END"
```

LISTING 2

```
2560 "POKE.LM":REM POKE LM FROM DATA
2570 Z=Q
2580 FO K=ITO 500:WAIT 0
2590 READ A$:CLS
2600 IF A$="END"WAIT: PRINT "SIZE:" ; Z-Q;" octets": RETURN
2620 PRINT A$ ;
2630 FOR J=ITO LEN A$-1STEP 2
2635 Z$=MID$ (A$,J,1)
2640 IF Z$="."OR Z$=" "LET J=J-1:NEXT J:NEXT K
2660 B=ASC Z$-48:C=(B-7*(B>9))*16
2670 B=ASC MID$ (A$,J+1,1)-48:C=C+B-7*(B>9)
2680 POKE Z,C:Z=Z+1:NEXT J:NEXT K
2720 REM
```



LISTING 3

```
1000 "COCHEZ":REM /SOUS ROUTINE DE COCHAGE
1010 C=&78BE
1030 Y$="3E5D6B776B5D3E":B=5
1040 FOR J=1 TO 16
1050 IF J=5LET C=&700E
1060 A=B+3:B=LEN   $(J)*6
1070 IF B<>0LET B=A+B+1:POKE C+J*16,A,B:NEXT J
1080 Z=J-1:WAIT 0:CLS:IF A-2>155BEEP 1:WAIT:PRINT A;" TROP LONG!":END
1090 GPRINT Y$;:GCURSOR 10:Y$="00000000":WAIT 0
1100 FOR J=1 TO Z:PRINT   $(J);:GPRINT Y$;:NEXT J:PRINT :BEEP 2
1120 FOR J=1 TO Z+1
1130 IF J<1LET J=1
1140 IF J>ZLET J=Z
1150 A=&78BE*(J<5)+&700E*(J>4)+J*16
1160 POKE &774E,PEEK A, PEEK (A+1)
1170 FOR A=1 TO 30
1180 CALL W:PRINT :FOR B=1TO 15:Z$=INKEY$: IF Z$=""NEXT B
1190 CALL W:PRINT:FOR B=1TO 15:Z$=INKEZY$: IF Z$=""NEXT B
1210 R=ASC Z$:IF R=130 R=43 RETURN
1220 IF R=8LET J=J-2:NEXT J:END
1230 IF R=12NEXT J
1240 NEXT A:J=0:RETURN
```

LISTING 4

```
2000 "Z":REM exemple 1
2010 REM Thème : codification des cartes d'un jeu de bridge (1 à 52)
2011 REM
2015 W=&7150:REM de la routiine L.M. (ici P$--S$)
2020 WAIT 0:PRINT "valeur de la carte ?";
2030 A$="AS":B$="R":C$="D":D$="V"
2040 FOR J=10 TO 2STEP -1:@$(J+3)=STR$ (12-J):NEXT J:@$(14)=""
2050 GOSUB "COCHEZ"
2060 B$="ROI":C$="DAME":D$="VALET":X$=@$(J):V=14-J
2080 A$="trèfle":B$="carreau":C$="cœur":D$="piq.":E$=""
2080 PRINT "sa couleur ?";
2100 GOSUB "COCHEZ"
2110 D$="Pique": V=V+13*(J-1):PRINT X$;"DE";@$(J);
2120 Z=PEEK &7875:PRINT "code"; V
2140 POKE &774E,Z+4,Z+30:CALL W
2150 WAIT: PRINT :GOTO "Z
2160 REM   ======================================
2170 REM
2200 "X":REM / Exemple 2
2215 W=&7150:REM   de la routine L.M. (ici P$--S$)
2220 WAIT 0:PRINT "DATE de MARIGNAN ?";
2230 A$="732":B$="1515":C$="1789":D$="1815":E$="1916";F$=""
2240 GOSUB "COCHEZ"
2250 WAIT :IF J=2PRINT "VRAI bravo !":END
2260 PRINT "FAUX encore une fois ENTER":CLS :GOTO "X"
```

DETOURNEMENT DU CLAVIER

Vous avez vu dans le bulletin numéro 5 une possibilité de réassigner le clavier de votre PC 1500. Aujourd'hui, nous allons vous présenter une méthode beaucoup plus puissante, le détournement du clavier :

Il existe en ROM une routine appelée par le moniteur qui se charge de la saisie au clavier, elle a pour adresse E2 43 hexa. Cette routine qui permet d'utiliser SHIFT, SMALL, DEF, etc., donne au retour dans l'accumulateur le code ASCII du caractère saisi. Le désassemblage de cette routine montre quelque chose de très intéressant :

E243	LD A, (79D4)	teste le flag 79D4
E246	CP A, 55	compare la valeur à 55
E248	JRZ, E2B7	saut en E2B7 si zéro

E2B7	LDX, (785B)	E2BD	SPV
E2B9	NOP	E2BE	JR NC, E2C2
E2BA	LDA, XL	E2C0	RPV
E2BB	SRA	E2C2	LD X,P
E2BC	DI		

Notez que le désassemblage ci-dessus est celui d'un ordinateur équipé d'une nouvelle ROM. Pour savoir quelle ROM vous avez, faites PEEK E2B9 ; si 56 (&38=NOP) s'affiche vous avez une nouvelle ROM, sinon cet article n'est pas pour vous...

Donc, si l'octet qui a pour adresse 79D4 vaut 55 hexa, on se branche à l'adresse indiquée par 785B, 785C, ceci sans passer par la routine de saisie (qui ne commence vraiment qu'en E24A). Il est ainsi possible d'écrire sa propre routine de saisie (un DRIVER), qui remplacera celle conçue initialement pour votre PC. Voici un exemple : Nous allons placer notre routine de 78C0 à 78DF (variables A$ et B$) :

78C0	OR(7B0E), 40	flag de répétition des touches
78C4	CALL E24A	on appelle la routine de saisie
78C7	CP A 0F	compare avec la touche OFF
78CB	JPE33F	si OFF alors on fait un auto power off

```
78CE PUSH A      sauve l'accumulateur
78D0 LD UL, 60   fréquence
78D2 LD XH,01
78D4 LD XL, 20   X = durée du BEEP
78D6 CALL E66F   BEEP 1, UL, X
78D9 POP A       récupère
                 l'accumulateur
78DB RET         retour au moniteur
```

La routine entrée, faites POKE & 785 B, &78, &C0 puis POKE &79D4, & 55. Vous disposez à présent d'un clavier à répétition, l'appui sur une touche est marquée par in petit BEEP et la touche OFF agit maintenant comme si la machine s'éteignait par un AUTO POWER OFF.

Attention ! Telle quelle, votre routine peut être effacée par un CLEAR et si cela se produit votre ordinateur va se "planter". Je vous conseille de placer cette routine à un autre endroit de votre choix (la routine est intégralement relogeable). Bien sûr, vos routines peuvent être plus développées, vous pouvez modifier le sens de certaines touches, protéger votre ordinateur, etc. Notre DRIVER ne fait qu'une vingtaine d'octets, il en existe qui font plusieurs kilo-octets comme le programme PC VISION (éditeur pleine page) qui n'est qu'un gigantesque driver !
Nous verrons une utilisation pratique d'un driver dans l'article sur les touches du MODE RESERVE.

PASCAL ABRIVARD

LIGNE BASIC EN ZONE RESERVE

Cette routine correspond à un besoin réel quand on écrit des programmes assez longs. Elle va permettre de concatener deux lignes de programme sans avoir à en retaper au moins une. Pour faire cette routine, on s'est servi d'une particularité bien utile. A chaque fois qu'une ligne de programme est affichée, l'adresse de son premier octet se trouve dans les octets 78A6 et &78A7. A partir de là il ne reste plus qu'à transférer dans la zone RESERVE en supprimant tout ce qui fait la particularité des lignes de programme :

- les deux octets du numéro de ligne
- l'octet de la longueur
- l'octet &0D à la fin de la ligne

A partir de ce moment, vous obtiendrez la réplique de la ligne en appuyant sur la touche correspondant au numéro de la réserve.
Pour rentrer le programme nous vous conseillons d'utiliser la routine «POKE.LM » donnée sur le listing 2 avec pour DATA ceux donnés sur le listing 1. Pour le lancer, faire DEF T vous aurez alors en mode réserve 1 sur la touche 3 (♯) CALL &7190C et sur la touche 1 (!) rien pour l'instant;
Placez-vous sur une ligne soit grâce à LIST soit grâce aux flèches, et appuyez sur ♯ en mode réserve 1. Alors appuyer sur la touche !, votre ligne s'affichera, vous pouvez en faire ce que vous voulez, par exemple la mettre à la suite d'une autre pour gagner de la place en mémoire.

P. HENRY

LISTING 1

```
2800 REM / Ligne BASIC en zone RESERVE
2900 "T":RESTORE "T":Q=&7190:GOSUB "POKE.LM"
2910 A=PEEK &7863*256+&56
2915 REM on va poker en début de la zone réserve:3 call&7190 1,0
2920 POKE A,3,241,138,38,55,49,57,48,64,1,44,0:END
2950 REM
3000 REM codes L.M. ligne PGM - zone réserve>
3001 REM
3005 DATA "CCA6.44.44.05.2A    ":REM  début.ligne——►Xr
3010 DATA "05.DF.DF.28.44      ":REM long.texte——►Uh
3015 DATA "5878.5A63.15.18     ":REM  début memory——►Yh
3020 DATA "5A56.6A6E.B501      ":REM Prépa recherche  code 01 en zone réserve
3025 DATA "17.8B04.54.8806     ":REM boucle recherche  code 01 en zone réserve
3030 DATA "E4.54.24.26.8301    ":REM y a-t-il encore assez de bytes en zone réserve
3035 DATA "E0.A4.2A            ":REM Uh——►UL
3040 DATA "F5.8803             ":REM boucle de transfert ligne basic——►réserve
3045 DATA "B500.1E.9A.01       ":REM zéro en fin de zone et RTN
3050 DATA "END"
```

LISTING 2

```
2560 "POKE.LM":REM /Poke LM from DATA
2570 Z=Q
2580 FOR K=1TO 500:WAIT 0
2590 READ A$:CLS
2600 IF A$="END"WAIT:PRINT"SIZE:";Z-Q;" octets":RETURN
2620 PRINT A$;
2630 FOR J=1TO LEN A$-1STEP 2
2635 Z$=MID$ (A$,J,1)
2640 IF Z$="."OR Z$=" "LET J=J-1:NEXT J:NEXT K
2660 B=ASC Z$-48:C=(B-7*(B>9))*16
2670 B=ASC MID$ (A$,J+1,1)-48:C=C+B-7*(B>9)
2680 POKE Z,C:Z=Z+1:NEXT J:NEXT K
2720 REM
```

LES INCONNUES DU MODE RESERVE

Dans l'article sur l'évaluateur du PC 1500 (bulletin numéro 8), on pouvait trouver une table pour les codes ASCII allant de 0 à 31. Cette table laissait apparaître des codes inconnus, et plus particulièrement ceux de code ASCII (F0) et 23 (F7).

Dans le désassemblage de la routine de traitement des touches du MODE RESERVE, on s'aperçoit que l'ordinateur accepte des codes allant de 10 hexa à 17 hexa, les codes 10 et 17 restant un mystère...

Ces codes correspondent à deux nouvelles touches RESERVE. Votre PC 1500 en possède donc 8 au total et pas seulement les 6 accessibles au clavier.

Regardons d'un peu plus près le mode RESERVE de votre ordinateur : lors de l'affectation d'une touche, l'ordinateur met un code qui caractérise la touche affectée en fonction du mode (I, II, III), suivi du texte affecté à la touche. La fin du mode RESERVE est marquée par un 0.

Les codes des touches sont donnés par la table suivante :

mode I		mode III	mode II	
F0 (DEF Z)	0	8	16	
F1 (!)	1	9	17	
F2(")	2	10	18	Notez
F3 ( )	3	11	19	l'inversion
F4($)	4	12	20	entre
F5(%)	5	13	21	les
F6(&)	6	14	22	modes
F7(DEF X))	7	15	23	II et III

Le problème pour utiliser les deux nouvelles touches RESERVE est de générer les codes ASCII 10 hexa et 17 hexa. On peut soit réassigner des touches du clavier avec les deux codes (cf bulletin numéro 5), soit utiliser un DRIVER.

Pour la seconde méthode on va changer sous driver la signification de <DEF> "Z" et de <DEF> "X", pour cela reprenons la routine donnée en exemple dans l'article précédent et complétons la :

78 DB	CP A, 9A	compare au code DEF Z
78DD	JR NZ, 78 E3	sauf si différent
78DF	LD A, 10	Si DEF Z, change en ASCII 10
78E1	JR 78E9	sauf à la fin
78E3	CP A, 98	compare au code DEF X
78E5	JR NZ, 78E9	sauf si différent
78E7	LD A, 17	si DEF X, change en ASCII 10
78E9	RET	retour au moniteur

Pour accéder à F0 faites sous mode réserve DEF Z, et DEF X pour F7. Tout fonctionne très bien en dehors du F0 en mode I, en effet son code caractéristique est 0, ce qui indique la fin de la réserve.

Vous voilà à présent avec 8* 3 = 24 touches à assigner !

PASCAL ABRIVARD

programmes

```
    1:REM ******************************
    2:REM *                            *
    3:REM *       MORPION 10*10        *
    4:REM *     PAR GIRONDOT marc      *
    5:REM *                            *
    6:REM ******************************
   25:WAIT 0:CLEAR :DIM A(99),B(99),V(
      21),B$(5):FOR J=0TO 4:READ V(J):
      NEXT J
   26:ON ERROR GOTO 30:COLOR 3:CSIZE 1
   30:FOR J=5TO 20STEP 5:READ V(J):
      NEXT J:READ B$(0),B$(1),B$(5)
   35:DATA .01,.03,.5,10,10000,.1,2,10
      0,1000000," ","O","X"
   47:GOTO 700
  200:ON ERROR GOTO 250:GRAPH :CSIZE 1
      :SORGN
  210:LPRINT "   0 1 2 3 4 5 6 7 8 9":
      GLCURSOR (0,-12)
  220:FOR J=0TO 9:SORGN :LPRINT STR$ J
      +"0";
  230:GLCURSOR (0,-12):NEXT J:GLCURSOR
      (14,-2):SORGN :FOR I=0TO 98STEP
      24:LINE (0,I)-(120,I)
  231:LINE (120,I+12)-(0,I+12):NEXT I:
      LINE (0,120)-(120,120)
  232:FOR I=0TO 98STEP 24:LINE (I,120)
      -(I,0)
  233:LINE (I+12,0)-(I+12,120):NEXT I:
      LINE (120,120)-(120,0)
  235:TEXT :CSIZE 1
  240:LF 6
  250:RETURN
  300:ON ERROR GOTO 350:GRAPH :
      GLCURSOR (0,170):SORGN :CSIZE 1:
      Y=INT (X/10):GLCURSOR (18+(X-Y*1
      0)*12,-Y*12)
  310:LPRINT B$(A(X));:GLCURSOR (0,-11
      0)
  340:TEXT :BEEP 1:CSIZE 1:LF 6
  350:RETURN
  700:CLS :PRINT "Commencez-vous ?";:
      GOSUB 9000:GOSUB 200:IF R$="O"
      GOTO 770
  750:GOSUB 1030
  760:IF F<>0GOTO 830
  770:WAIT :GOSUB 2000
  780:IF F=0THEN 750
  830:WAIT 0:BEEP 1:IF F=1PRINT "J ai
      gagne en";:GOTO 840
  832:IF F=-1PRINT "J ai perdu en";:
      GOTO 840
  834:PRINT "Partie nulle apres";
  840:WAIT :PRINT N;" coups":END
 1030:S=-1:WAIT 0:CLS :PRINT "Je jou
      e . . . ":WAIT
 1050:FOR I=0TO 99:B(I)=0:NEXT I
 1090:C=1:J1=0:J2=5:K2=90:GOSUB 1500
 1100:C=9:J1=4:J2=9:K2=50:GOSUB 1500
 1110:C=10:J1=0:GOSUB 1500
 1120:C=11:J2=5:GOSUB 1500
 1170:S=-1:FOR J=0TO 99
 1220:IF (B(J)>SOR (B(J)=SAND RND (0
      )>.8))AND A(J)=0LET C=J:S=B(C)
 1230:NEXT J
 1240:R$=STR$ C:IF LEN R$=1LET R$="0
      "+R$
 1245:BEEP 1:CURSOR 8:PRINT "en ";R$
      :A(C)=5
 1250:X=C:GOSUB 300:N=N+1:IF N=100OR
      (S=0AND N>1)LET F=2
 1290:IF S>=V(20)LET F=1:X=C:GOSUB 1
      0000
 1300:RETURN
 1500:C4=4*C
 1530:FOR J=J1TO J2:FOR K=0TO K2STEP
      10:A=J+K:B=A+C4:Z=0
 1570:FOR P=ATO BSTEP C:Z=Z+A(P):
      NEXT P
 1580:Q=V(Z):IF Q<>0FOR P=ATO BSTEP
      C:B(P)=B(P)+Q:NEXT P
 1600:NEXT K
 1610:NEXT J
 1620:RETURN
 2000:CLS :INPUT "Votre coup ";A
 2030:IF A<0OR A>99OR A<>INT AOR A(A
      )<>0THEN 2000
 2040:R$=STR$ A:IF LEN R$=1LET R$="0
      "+R$
 2060:A(A)=1:N=N+1:X=A:GOSUB 300
 2100:IF N=100LET F=2
 2110:IF B(A)>=V(4)LET F=-1:X=A:
      GOSUB 10000
 2220:RETURN
 9000:R$=INKEY$ :IF R$=""THEN 9000
 9010:IF R$<>"O"AND R$<>"N"PRINT "O
      ou N":GOTO 9000
 9020:BEEP 1:RETURN
10000:J=1:GOSUB 10010:FOR J=9TO 11:
      GOSUB 10010:NEXT J:RETURN
10010:Y=X
10020:ON ERROR GOTO 10059:IF (Y+1)/1
      0=INT ((Y+1)/10)AND (J=1OR J=1
      1)GOTO 10060
10030:IF Y/10=INT (Y/10)AND J=9GOTO
      10060
10040:IF A(X)<>A(Y+J)GOTO 10060
10050:Y=Y+J:GOTO 10020
10059:Y=Y-J
10060:U=Y:W=0
10061:W=W+1:IF W=5GOSUB 10070:RETURN
10062:ON ERROR GOTO 10069:IF (U+1)/1
      0=INT ((U+1)/10)AND J=9LET
10063:IF U/10=INT (U/10)AND (J=1OR J
      =11)LET
10064:IF A(X)<>A(U-J)LET
10065:U=U-J:GOTO 10061
10069:RETURN
10070:ON ERROR GOTO 10100
10080:GRAPH :CSIZE 1:GLCURSOR (0,170
      ):SORGN :A=INT (Y/10):B=INT (U
      /10)
10090:LINE (20+(Y-A*10)*12,3-A*12)-(
      20+(U-B*10)*12,3-B*12),0,1
10095:GLCURSOR (0,-110):BEEP 1:TEXT
      :CSIZE 1:LF 6
10100:RETURN
```

Taper RUN et répondre par QO ou Y pour savoir si vous commencer. Pour jouer, donner vos coordonnées sous la forme : ligne en dizaine et colonnes en unit.

LANGAGE MACHINE
4e PARTIE

DECALAGE VERS A GAUCHE ET A DROITE :
Le microprocesseur du PC 1500 dispose d'instructions qui sont indispensables à tous calculs, ce sont les décalages. Il en existe 4 qui travaillent sur des entiers de 8 bits :
SLA : décalage de 1 bit vers la gauche sans carry (BIT0 = 0)
SRA : décalage de 1 bit vers la droit sans carry (BIT7 = 0)
RLA : décalage de 1 bit vers la gauche avec carry (BIT0 = C)
RRA : décalage de 1 bit vers la droite avec carry (BIT7 = C)
Les décalages correspondent à une multiplication ou à une division par 2 de l'accumulateur, ce qui est parfait pour tous les calculs sur des entiers de 8 ou 16 bits.
Nous allons voir à présent des exemples d'utilisation des décalages en écrivant une routine de multiplication et de division sur les entiers de 8 bits.

LA MULTIPLICATION
La multiplication en binaire s'écrit exactement comme pour la base 10, voyons ceci sur un exemple, multiplions 5 par 6 :

```
(5)         101
(6)       x 110
          -----
            000
           1010
          10100
          -----
(30)      11110
```

Comme on le voit, une multiplicaiton se construit par une suite de décalages et d'additions.
Pour un programme multipliant deux nombres de 8 bits, par exemple XL*UH avec le résultat dans Y, les étapes seront :
1) mettre 0 dans le résultat
2) décalage à droite du multiplicande
3) si le bit 0 du multiplicateur vaut 0 allez en 4
4) additionner le multiplicande au résultat
5) décaler le multiplicateur vers la gauche
6) recommencer au 2 tant que le multiplicateur n'est pas nul

Notez que l'on peut, plutôt que de comparer le multiplicateur à zéro, boucler 8 fois de suite. En effet au bout de 8 décalages vers la droite on est sur qu'un entier 8 bits est nul mais on va perdre du temps pour rien si le multiplicateur est petit.

Le programme est le suivant :

```
DEBUT    LDI XH,00     XH sert de tampon pour les décalages de UL
         LDI YL,00
         LDI YH,00     mise à zéro du registre Y
BOUCLE   RCF
         LDA UH        charge l'accumulateur avec le multiplicateur
         RRA           rotation à droite, le bit 0 de UH passe dans le carry
         STA UH        UH = UH/2
         BCR nonadd    si ce bit est à zéro, pas d'addition
         LDA XL        si le carry vaut 1 il n'y a pas de saut et on additionne
         ADD Y,A       le registre X au registre Y.
         LDA XH        l'absence d'opération directe sur des nombres de 16 bits
         RCF           oblige à écrire : YL = YL + XL
         ADC A,YH      puis : YH = YH + XH avec retenue
         STA YH
NOADD    LDA XL        on multiplie le registre X par 2, ici aussi on est obligé
         SLA           de décomposer l'opération en deux
         STA XL
         LDA XH
         RLA
         STA XH
         LDA UH
         BZR boucle
FIN      RET           on réitère la boucle principale.
```

LA DIVISION :
Même si le programme semble un peu plus compliqué, l'algotithme reste un peu près le même, mais au lieu de faire des additions, on soustrait le diviseur au dividende.
Un programme de division 8 bits/7 bits (YL/XL avec au retour, le quotien dans XH et le reste dans YH) pourra s'écrire :

```
         LDI YH,00     mise à zéro du tampon pour les décalages de YL
         LDI XH,00     mise à zéro du dividende
         LDI UL,07     on exécute 8 fois la boucle principale
boucle   LDA XH
         SLA           on décale XH vers la gauche
         STA XH
         LDA YL
         SLA           on décale le registre Y vers la gauche
         STA YL        ici aussi, il faut décomposer l'opération en
         LDA YH        deux temps.
         RLA
         STA YH
         SCF           met le carry à zéro avant la soustraction
         SCB A, XL     soustrait le diviseur à l'accumulateur (A-YH)
         BCR inf       saut si A est inférieur à XL
         STA YH        sinon YH = YH-XL
         INC XH        incrémente XH
inf      LOP boucle    retourne à la boucle principale
         RET           fin
```



LES NOMBRES PREMIERS :

Pour illuster un peu les possibilités du langage machine, nous allons écrire un programme de recherche des nombres premiers en utilisant les routines précédentes.
Notre programme utilisera l'algorithme très simple qui consiste à diviser un nombre par chacun des termes de la suite : 2,3,5,7,9..., n avec n inférieur à la racine du nombre. Si un élément de la suite divise notre nombre c'est que celui-ci n'est pas premier.
Bien sûr, notre division ne permettra pas de traiter des nombres de plus de 255, mais rien ne vous empêche d'écrire, en reprenant le même algorithme, une division 16/15 bits ou même 32/31 bits.
Le passage des paramètres va se faire par l'octet 78C0 pour le nombre à tester et 78C1 pour le flag de retour, si ce drapeau est à 1 c'est que le nombre est premier.
L'algorithme va s'écrire :
lecture du nombre
si pair alors pas premier
i = 1
i = i + 2
calcul n/i
test le reste de la division
si reste = 0, flag = 0 et fin
sinon
le diviseur est-il inférieur au quotient ?
si oui alors boucle
sinon
flag = 1 et fin
fin
La routine en langage machine est dans les DATA du programme basic qui suit. La routine est implantée de 78C0 à 78F0. Pour utiliser le programme, faites DEF N et entrez le nombre à tester (de 1 à 255) : la réponse arrive instantanément.

PASCAL ABRIVARD

——— Programme page 48 ———



RENUMBER V1.0 POUR LES PC1245, 1250, 1251, 1255

Ce programme de renumérotation est un utilitaire bien pratique. Il permet de renuméroter un programme à partir d'une ligne voulue, avec un incrément spécifié au moment de la renumérotation. Ce programme se charge de gérer les GOTO, GOSUB et autres instructions de saut.

Destiné à vous montrer l'efficacité du L.M. ce programme n'a pas été complètement optimisé. Pour commencer vous pourrez essayer de le réduire d'environ 15 octets, opération facile ne nécessitant pas un trop gros travail. Dans sa version actuelle le RENUMBER occupe 360 octets. Pour charger ce programme en mémoire sur un PC-1251 : utiliser le programme de désassembleur fournit dans le dernier numéro et utiliser l'option C (rentrer tel quel le programme annexe 1). Sur un PC-1245 : rentrer le programme à la main (POKE commence en C000), pour exemple : &C460 (adresse du début du programme possible en donnant cette adresse). Si vous le désirez ce programme peut être implanté ailleurs en mémoire, il suffit de le mettre où l'on veut puis d'exécuter le programme RELOGEUR REN en indiquant l'adresse où l'on a mis le RENUMBER (programme RELOGEUR REN annexe 2).

Pour faire fonctionner le programme, on met dans la variable A le nouveau numéro de ligne de départ du programme, c'est donc une valeur comprise entre 0 et 1000 exclus. On met dans la variable B l'incrément entre 2 numéros de ligne ; c'est une valeur qui devrait être comprise entre 0 et 100 exclus. Puis, on fait CALL adresse début du programme soit ici CAL & BF00.

ATTENTION : Ce programme ne teste pas toutes les bêtises que vous pourriez faire. Ainsi un A ou B alphanumérique ou un envoi à un numéro de ligne inexistant provoque un message d'erreur, mais le programme sera peut-être en pagaille. De plus un A ou un B négatif provoque souvent un ALL RESET.
N'essayez pas de faire non plus de combinaison A et B qui mènerait à une numérotation qui dépasserait 999.
Ce programme renumérote toutes les structures GOTO THEN GOSUB RESTORE simple ou répétitives (ON A GOTO...,...,...).
Les labels ou les expressions calculés seront laissés tel quel (voir 1ère ligne de l'exemple annexe 3).
Cas où la renumérotation ne s'effectuera pas : structure IF THEN GOTO remplacez cette structure par IF GOTO ou IF THEN.
Pour vous permettre de mieux comprendre ce programme vous en trouverez en annexe 4 l'organigramme, et en annexe 5 le listing commenté.
De plus voici dans l'ordre d'apparition dans le programme la signification des CAL ROM CPU.

CAL	SIGNIFICATION	EXPLICATIONS
118F	X < - START BASIC	met dans le registre X le contenu de C6E3, C6E4
1200	[(1C), (1D)]< -X	sauve en Xen[(1C), (1D)] X_H -> (1D) XL -> (1C)
15A3	Interprétation numérique	Interprétation d'un numéro pur si X pointe sur un numéro, il sera interprété et stocké au format variable de 20 à 27 et X pointera sur le dernier caractère de la valeur interprétée. Si on pointe sur un label ou une expression numérique il ne se passera rien.
1ACF	< - > [(1C),(1D)]	échange X_L et (1C) X_H et (1D)
1177	Y< - X	copie X dans Y
1195	< - > [(OC),(OD)]	échange X_L et (OC) H_H et (OD)
1192	X< - [(DP),(DP+1)]	met dans X ce sur quoi pointe DP
1172	X< - Y	copie Y dans X
11F9	Y<≷ adr. buffer - 1	met dans Y < -C7CF adresse du buffer disp - 1
11AF	Affich.	routine d'affichage du contenu du buffer
11E0	DISP ON	Allumage de l'affichage
1D23	INP KEY	attend jusqu'à la pression d'un caractère

ANNEXE 1

```
DUMP RENUMBER
BF00 F18F 2467 FF38 D867
BF08 D738 0F67 D838 0B67
BF10 DF38 0767 9238 032D
BF18 16F2 00F5 A324 78C0
BF20 003A 0D24 78C0 002B
BF28 0567 1339 132D 2C78
BF30 C00F 78C0 2124 67FF
BF38 3897 67E0 3B08 A2C7
BF40 288A 2450 C728 85B0
BF48 5964 4FB4 DBB1 5964
BF50 F065 0458 B5DB B159
BF58 640F 6540 B6DB B723
BF60 DB03 02B4 6340 280D
BF68 0003 0201 34B7 1D2F
BF70 03C3 2D10 DA58 A145
BF78 8058 DBFA CF3A 0828
BF80 3124 B419 2D62 F177
BF88 F195 10C6 E3F1 9202
BF90 FF26 F177 A123 DB45
BF98 DB58 0300 8614 02FF
BFA0 0726 78C0 5C25 67FF
BFA8 3804 272D 07F1 952D
BFB0 2FF1 77F1 95F1 72A1
BFB8 5958 0300 8414 2426
BFC0 67FF 2905 78C0 5CF1
BFC8 952D 4978 BFF5 2D9A
BFD0 F1F9 0296 2623 26F1
BFD8 AFF1 E0FD 2337 78C0
BFE0 2124 67FF 380F 67E0
BFE8 3B08 B053 5004 5378
BFF0 BFF5 2D12 3734 B259
BFF8 5100 000C 23C4 5B37
C000 671B 380A 671D 3806
C008 6700 3802 37D0 37A1
C010 0220 DB45 DB58 3400
C018 01A2 1C2F 05A2 61E0
C020 37B8 1321 0002 0810
C028 C690 A000 0F18 A063
C030 F538 2578 C00F A359
C038 B2DB A013 2800 0708
C040 A063 F538 1378 C00F
C048 B013 2200 0108 A113
C050 3800 0208 F18F 375B
C058 5B79 BFD0 10C6 E386
C060 5952 8759 11E4 5237
```

ANNEXE 2

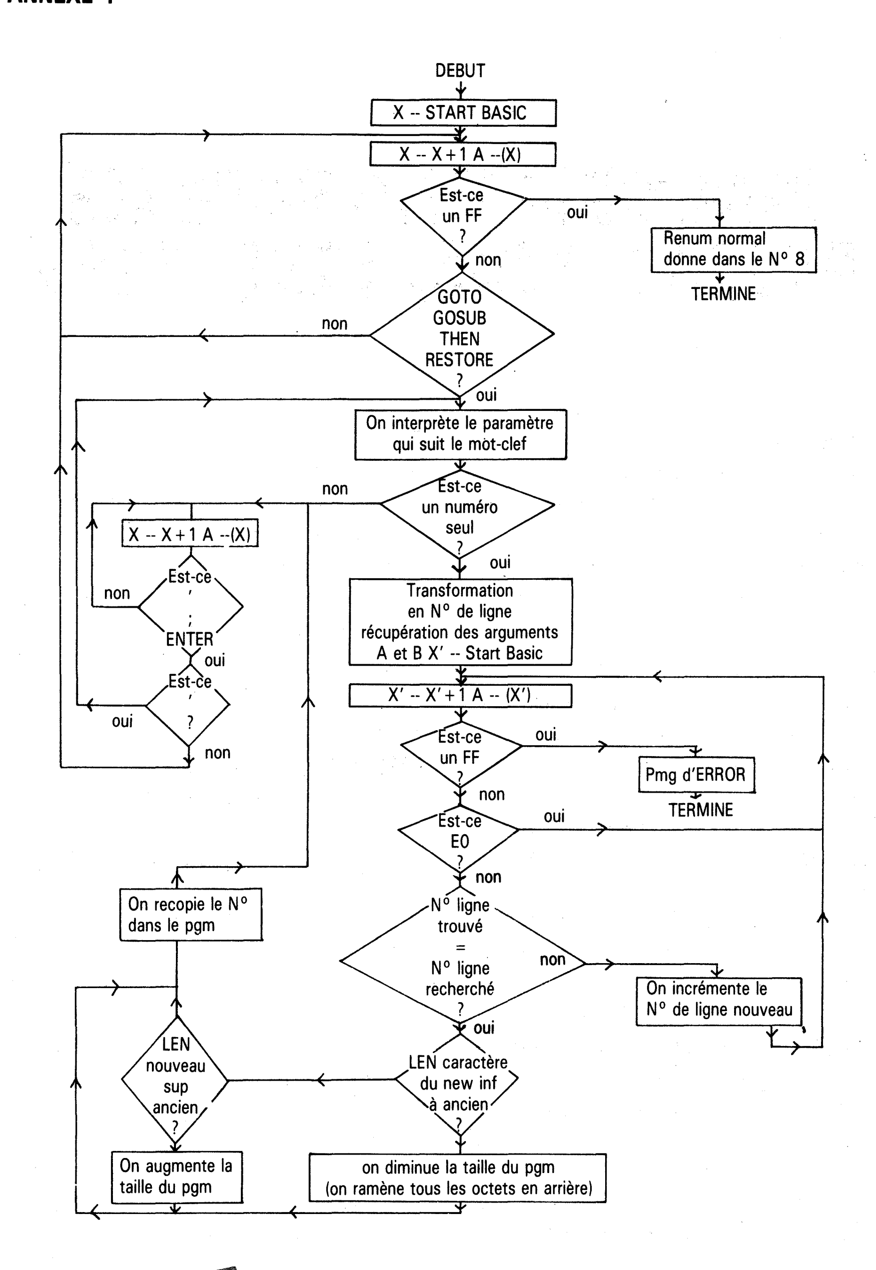
```
RELOGEUR REN
10:INPUT "@DR:";A
20:FOR I=1 TO 12: READ
   B,C:D= INT ((A+C)/25
   6): POKE A+B,D,(A+C)
   -256D: NEXT I
30:END
40:DATA 31,256,37,256,4
   8,271,51,289,163,348
   ,197,348,204,245,223
   ,289,240,245,308
50:DATA 271,326,271,346
   ,208
```

ANNEXE 3 Exemple

```
LLIST
1:ON A GOTO 2,3,"#",5,
  2*B,6
2:RESTORE 1
3:IF A=2 THEN 4
4:GOSUB "#"
5:"#" GOSUB 6
6:RETURN
```

```
A=80  B=10  CALL &BF00
 80:ON A GOTO 90,100,"#"
    ,120,2*B,130
 90:RESTORE 80
100:IF A=2 THEN 110
110:GOSUB "#"
120:"#" GOSUB 130
130:RETURN
A=10  B=5  CALL &BF00
10:ON A GOTO 15,20,"#",
   30,2*B,35
15:RESTORE 10
20:IF A=2 THEN 25
25:GOSUB "#"
30:"#" GOSUB 35
35:RETURN
```

ANNEXE 4



ANNEXE 5

BF00 F1 CAL 118F	On met dans X l'adr où commence la zone pgm
BF02 24 IXL	
BF03 67 CPIA FF	On regarde s'il y a un FF
BF05 38 JRZP D8 (BFDE)	Si oui fin de scrutation de cette zone, on va en BFDE
BF07 67 CPIA D7	On regarde si on a GOTO
BF09 38 JRZP 0F (BF19)	Si oui on va BF19
BF0B 67 CPIA D8	On regarde si on a RESTORE
BF0D 38 JRZP 0B (BF19)	Si oui on va en BF19
BF0F 67 CPIA DF	On regarde si on a THEN
BF11 38 JRZP 07 (BF19)	Si oui on va en BF19
BF13 67 CPIA 92	
BF15 38 JRZP 03 (BF19)	
BF17 2D JRM 16 (BF02)	Autrement, on retourne en BF02
BF19 F2 CAL 1200	Sauve X en ([1C], [1D])
BF1B F5 CAL 15A3	On interprète le paramètre suivant les fcts citées
BF1D 24 IXL	On charge dans A le caract. qui suit ce qui a été interprété
BF1E 78 CALL C000	On regarde si on a une fin de zone (:, ou ENTER)
BF21 3A JRCP 0D (BF2F)	Si l'interprété est bon, on va en BF2F
BF23 24 IXL	
BF24 78 CALL C000	
BF27 2B JRNCM 05 (BF23)	On boucle jusqu'à l'arrivée en fin de zone de traitement)
BF29 67 CPIA 1B	On regarde si on a une « , »
BF2B 39 JRZM 13 (BF19)	Si oui on recommence une interprétation de ce qui suit
BF2D 2D JRM 2C (BF02)	Autrement on va à la recherche d'un mot clef
BF2F 78 CALL C00F	Récupération d'un argument sous forme d'un N° ligne
BF32 78 CALL C021	Récupération des arguments stockés dans A et B
BF35 24 IXL	On regarde le caractère suivant
BF36 67 CPIA FF	A-t-on fini de scruter la zone pgm ?
BF38 38 JRZP 97 (BFD0)	Si oui on va en BFD0, car le N° de ligne recherché n'existe pas
BF3A 67 CPIA E0	A-t-on un numéro de ligne ?
BF3C 3B JRCM 08 (BF35)	Non on continue, on va en BF35
BF3E A2 LP22	
BF3F C7 CPMA	
BF40 28 JRNZP 8A (BFCB)	On regarde si les numéros de ligne sont identiques
BF42 24 IXL	(numéro trouvé = numéro recherché)
BF43 50 INCP	Si identiques, on continue
BF44 C7 CPMA	Si différents, on va en BFCB
BF45 28 JRNZP 85 (BFCB)	
BF47 30 LP30	
BF48 59 LDM	
BF49 64 ANIA 4F	
BF4B B4 LP34	Convertit le numéro de ligne stocké
BF4C DB EXAM	en [30], [31] en binaire, en un numéro en
BF4D B1 LP31	code caractère par chiffre de [34] à [37]
BF4E 59 LDM	
BF4F 64 ANIA F0	
BF51 65 ORIA 04	
BF53 58 SWP	
BF54 B5 LP35	
BF55 DB EXAM	
BF56 B1 LP31	
BF57 59 LDM	
BF58 64 ANIA 0F	
BF5A 65 ORIA 40	
BF5C B6 LP36	
BF5D DB EXAM	
BF5E B7 LP37	
BF5F 23 LDS	

```
BF60 DB   EXAM
BF61 03   LIB    02
BF63 B4   LP34
BF64 63   CPIM   40
BF66 28   JRNZP  0D (BF74)
BF68 00   LII    03
BF6A 02   LIA    01
BF6C 34   DO
BF6D B7   LP37
BF6E 1D   SLW
BF6F 2F   LOOP   03 (BF6D)
BF71 C3   DECB
BF72 2D   JRM    10 (BF63)
BF74 DA   EXAB
BF75 58   SWP
BF76 A1   LP21
BF77 45   SBM
BF78 80   LP00
BF79 58   SWP
BF7A DB   EXAM
BF7B FA   CAL 1ACF
BF7D 3A   JRCP   08 (BF86)
BF7F 28   JRNZP  31 (BFB1)
BF81 24   IXL
BF82 B4   LP34
BF83 19   EXWD
BF84 2D   JRM    62 (BF23)
BF86 F1   CAL 1177
BF88 F1   CAL 1195
BF8A 10   LIDP   C6E3
BF8D F1   CAL 1192
BF8F 02   LIA    FF
BF91 26   IYS
BF92 F1   CAL 1177
BF94 A1   LP21
BF95 23   LDS
BF96 DB   EXAM
BF97 45   SBM
BF98 DB   EXAM
BF99 58   SWP
BF9A 03   LIB    00
BF9C 86   LP06
BF9D 14   ADB
BF9E 02   LIA    FF
BFA0 07   DY
BFA1 26   IYS
BFA2 78   CALL   C05C
BFA5 25   DXL
BFA6 67   CPIA   FF
BFA8 38   JRZP   04 (BFAD)
BFAA 27   DYS
BFAB 2D   JRM    07 (BFA5)
BFAD F1   CAL 1195
BFAF 2D   JRM    2F (BF81)
BFB1 F1   CAL 1177
BFB3 F1   CAL 1195
BFB5 F1   CAL 1172
BFB7 A1   LP21
BFB8 59   LDM
BFB9 58   SWP
BFBA 03   LIB    00
BFBC 84   LP04
BFBD 14   ADB
```

B ⟵ 2 nombre maximum de décalage
P ⟵ 34 on pointe sur le 1er caract. du nbre à mettre ds le pgm
Si c'est un 0, on va en BF74

On décale le nombre qui est de [34] à [37]
d'un octet vers la gauche
On décrémente B (le compteur de décalage)
et on retourne en BF63 tester le 1er
caractère de ce nombre

On sauve B dans A
On échange A5-8 et A1-4 (car l'expos est ds le P F)
On pointe sur l'exposant du nombre recherché
On les soustrait
On pointe sur I
On remet A dans un sens normal
On met dans I le résultat de la diff. des 2 exposants
On récupère X pointant sur le mot clef
Si carry, alors nouveau N° plus petit que l'ancien
Nouveau N° plus grand que l'ancien
On pointe derrière le mot clef
On pointe sur le nombre à copier
On copie ce nombre dans la zone pgm
Et on recommence tout

On diminue la mémoire pgm
car le nouveau numéro de ligne
est plus petit que l'ancien en
nombre de caractères

On augmente la mémoire pgm
car le nouveau numéro de ligne
est plus grand que l'ancien en
nombre de caractères

```
BFBE 24   IXL
BFBF 26   IYS
BFC0 67   CPIA  FF
BFC2 29   JRNZM 05 (BFBE)
BFC4 78   CALL  C05C
BFC7 F1   CAL 1195
BFC9 2D   JRM   49 (BF81)
BFCB 78   CALL  BFF5            On incrémente le nouveau N° de ligne
BFCE 2D   JRM   9A (BF35)       Et on retourne à la recherche du N° suivant à comparer
BFD0 F1   CAL 11F9
BFD2 02   LIA   96              Routine d'ERREUR
BFD4 26   IYS                   Affiche le mot ERREUR si on a un
BFD5 23   LDS                   numéro de ligne inexistant ou une
BFD6 26   IYS                   variable A ou B alphanumérique
BFD7 F1   CAL 11AF
BFD9 F1   CAL 11E0
BFDB FD   CAL 1D23
BFDD 37   RTN
BFDE 78   CALL  C021            Récupération des arguments
BFE1 24   IXL                   On regarde le caractère suivant
BFE2 67   CPIA  FF              Est-on en Fin de pgm ?
BFE4 38   JRZP  0F (BFF4)       Si oui on va en BFF4
BFE6 67   CPIA  E0              A-t-on un numéro de ligne ?
BFE8 3B   JRCM  08 (BFE1)       Non, on va en BFE1 on continue la scrutation
BFEA B0   LP30
BFEB 53   MVDM                  On copie le nouveau numéro de ligne
BFEC 50   INCP                  en mémoire pgm
BFED 04   IX
BFEE 53   MVDM
BFEF 78   CALL  BFF5            On incrémente le nouveau numéro de ligne
BFF2 2D   JRM   12 (BFE1)       On recommence
BFF4 37   RTN
BFF5 34   PUSH
BFF6 B2   LP32
BFF7 59   LDM                   Incrémente le nouveau numéro
BFF8 51   DECP                  de ligne de l'incrément spécifié
BFF9 00   LII   00
BFFB 0C   ADN
BFFC 23   LDS
BFFD C4   ADCM
BFFE 5B   POP
BFFF 37   RTN
C000 67   CPIA  1B              A-t-on une « , » ?
C002 38   JRZP  0A (C00D)       Si oui, on va en C00D (C     1 et on revient)
C004 67   CPIA  1D              A-t-on un « , » ?
C006 38   JRZP  06 (C00D)       Si oui, on va en C00D (idem)
C008 67   CPIA  00              A-t-on « ENTER » ?
C00A 38   JRZP  02 (C00D)       Si oui, on va en C00D (idem)
C00C 37   RTN                   On revient avec C     0
C00D D0   SC                    C     1
C00E 37   RTN                   On revient
C00F A1   LP21
C010 02   LIA   20              Le CAL 15A3 a interprété ce qui se trouvait
C012 DB   EXAM                  derrière le mot clef. Si l'argument était purement numérique, alors il a été stocké
C013 45   SBM                   dans la zone mémoire CPU allant de [20] à [27]
C014 DB   EXAM                  Ce sous-pgm se charge de le transformer en
C015 58   SWP                   une structure numéro de ligne : Exxx
C016 34   DO                    Ce sous-pgm sert aussi, lors de la saisie des
C017 00   LII   01              variables A et B
C019 A2   LP22
C01A 1C   SRW
C01B 2F   LOOP  05 (C017)
C01D A2   LP22
```

```
C01E 61   ORIM   E0
C020 37   RTN
C021 B8   LP38
C022 13   LIQ    21
C024 00   LII    02
C026 08   MVW
C027 10   LIDP   C690
C02A A0   LP20
C02B 00   LII    0F
C02D 18   MVWD
C02E A0   LP20
C02F 63   CPIM   F5
C031 38   JRZP   25 (C057)
C033 78   CALL   C00F
C036 A3   LP23
C037 59   LDM
C038 B2   LP32
C039 DB   EXAM
C03A A0   LP20
C03B 13   LIQ    28
C03D 00   LII    07
C03F 08   MVW
C040 A0   LP20
C041 63   CPIM   F5
C043 38   JRZP   13 (C057)
C045 78   CALL   C00F
C048 30   LP30
C049 13   LIQ    22
C04B 00   LII    01
C04D 08   MVW
C04E A1   LP21
C04F 13   LIQ    38
C051 00   LII    02
C053 08   MVW
C054 F1   CAL 118F
C056 37   RTN
C057 5B   POP
C058 5B   POP
C059 79   JP     BFD0
C05C 10   LIDP   C6E3
C05F 86   LP06
C060 59   LDM
C061 52   STD
C062 87   LP07
C063 59   LDM
C064 11   LIDL   E4
C066 52   STD
C067 37   RTN
```

Récupèration des arguments
On sauve le numéro de ligne lu
en [38], [39] avec l'exposant

DP pointe sur B
On copie B dans la zone CPU allant de [20] à [27]
et A[28] à [2F]

On regarde si B est alphanumérique
Si oui erreur, on va en C057.

On transforme ce qui est en B en un N° de ligne

On récupère ce qui est en [23] et on le sauve
en [32] : C'est la valeur d'incrément des lignes

Puis on met A dans la zone allant de (20) à (27)

On regarde si A est alphanumérique, si oui
erreur et on va en C057

On transforme A en un N° de ligne

On copie A (N° de ligne)
en (30) et (31)

On récupère en (21) (22) le N° de ligne à
rechercher, avec l'exposant

On recharge dans X l'adr du début de la zone pgm
On revient

On récupère l'adresse de retour du sous-
pgm d'erreur car A ou B était alpha

Réinitialise le pointeur de fin de pgm

DISP

Voici un petit programme en LM ESR-H sur PC 1251/45 qui je pense vous sera bien utile. En effet, il se propose d'afficher en les décalant lors d'un PRINT, tous les caractères tronqués jusqu'alors.

Il est relocalisable et occupe 128 octets. Et comme vous pourrez le constater, il est doté de perfectionnements agréables grâce à la gestion d'au plus 4 touches. Afin d'entrer dans le vif du sujet, je ne vous rappellerai pas la procédure avec ou sans le moniteur du n° 8, pour entrer un pgm LM dans le PC. Ou sa gestion sur K7 à l'aide des commandes CS. M et CLO. M, maintes fois décrites.

Le pgm tient compte de tous les cas de figures : PRINT = LPRINT, WAIT avec ou sans opérande. Il permet le contrôle du défilement par appui de touche et de la vitesse par WAIT n: 0<n<65535.

On peut donc facilement régler la vitesse du décalage, le stopper, l'accélérer, Breaker tout le pgm, ou passer à la suite du pgm basic principal avant la fin du défilement. Si on a moins de 25 caractères, pas de disp possible, et l'on peut aussi défiler pas à pas dans le sens d'un WAIT.

Seul a été délibérément omis le décalage inverse, c'est-à-dire de la gauche vers la droite, pour gagner des octets. Si le cœur vous en dit...

Avec les exemples donnés, vous avez pu observer la syntaxe. Je conseille fortement l'emploi du WAIT 0 pour éviter un tressautement lors du PRINT, le CALL venant après. Le passage des paramètres de vitesse est transmis par le second WAIT. Je signale aussi que la touche CL est inopérante dans tous les cas de figure sauf ruse vicieuse. La plupart des touches de contrôle étant à répétition automatique et instantanée. dernière recommandation : poussez un peu le contraste.

Et maintenant, ouvrez les yeux et à vos PC.

J.F. V.

Résumons les trois cas généraux :

1. PRINT d'au plus 80 caractères et au moins 25, avec WAIT (sans n)
   BRK: arrêt du LM et du basic avec message BREAK IN
   ENTER: arrêt du LM avec exécution de la suite du pgm basic.
   ▶ : défilement pas à pas avec autorépétition

Exemple :
10: WAIT 0: PRINT 80 car max: WAIT: CALL adr LM

2. PRINT d'au plus 80 car et au moins 25, avec WAIT n
   BRK: arrêt du LM et du basic avec message
   ENTER: arrêt du LM, exécution de la suite du basic
   ▶ : défilement très rapide
   SPC : arrêt momentané du décalage ; tant que la touche est enfoncée.

Exemple :
10: WAIT 0 : PRINT 80 car max: WAIT n: CALL adr LM

3. PRINT de moins de 25 caractères avec ou sans WAIT (avec ou sans n)
   BRK: arrêt LM et basic ; message
   ENTER: arrêt du LM et suite du pgm basic.

Exemple :
10: WAIT 0: PRINT moins 25 car: WAIT(n): CALL adr LM
l'instruction WAIT est dans ce cas facultative (et le pgm aussi si le LEN de l'affichage est connu).

```
DISP N
(C) JFV

BF00 10  LIDP  C6D7
BF03 D6  TSID  08              teste si on a PRINT = LPRINT
BF05 28  JRNZP 79 (BF7F)       Si oui RTN
BF07 11  LIDL  E7
BF09 57  LDD
BF0A 67  CPIA  19              regarde si on a moins de 25 caractères
BF0C 2A  JRNCP 0F (BF1C)       sinon exécute un disp n
BF0E F1  CAL 11E0              LCD ON
BF10 FF  CAL 1F44              scrutation du clavier
BF12 67  CPIA  07              texte si BRK
BF14 38  JRZP  65 (BF7A)       SI OUI RTN avec (BREAK IN)
BF16 67  CPIA  00              teste si ENTER
BF18 38  JRZP  64 (BF7D)       si oui LCD OFF ; RTN
BF1A 2D  JRM   0B (BF10)       si aucune de ces touches : Boucle
BF1C F1  CAL 119A              charge dans X adr-1 buffer out
BF1E 24  IXL                   incrémente
BF1F A0  LP20                  charge les 24 registres [20] à [37]
BF20 00  LII   17              avec le contenu du buffer à
BF22 18  MVWD                  partir de l'adr pointée dans X
BF23 57  LDD
```

```
BF24 67  CPIA   00
BF26 38  JRZP   56 (BF7D)
BF28 F2  CAL 1200
BF2A 78  CALL   400C
BF2D F1  CAL 11E0
BF2F FF  CAL 1F44
BF31 67  CPIA   07
BF33 38  JRZP   46 (BF7A)
BF35 67  CPIA   00
BF37 38  JRZP   45 (BF7D)
BF39 67  CPIA   11
BF3B 38  JRZP   08 (BF44)
BF3D 10  LIDP   C6D8
BF40 D6  TSID   04
BF42 38  JRZP   0F (BF52)
BF44 34  PUSH
BF45 88  LP08
BF46 03  LIB    FF
BF48 56  READ
BF49 F1  CAL 11EA
BF4B 5B  POP
BF4C 67  CPIA   0E
BF4E 38  JRZP   16 (BF65)
BF50 2D  JRM    22 (BF2F)
BF52 67  CPIA   0E
BF54 38  JRZP   21 (BF76)
BF56 23  LDS
BF57 03  LIB    00
BF59 88  LP08
BF5A F1  CAL 11EA
BF5C 10  LIDP   C6E5
BF5F 82  LP02
BF60 1A  MVBD
BF61 88  LP08
BF62 15  SBB
BF63 38  JRZP   19 (BF7D)
BF65 03  LIB    19
BF67 4E  WAIT   5F
BF69 6B  TEST   08
BF6B 28  JRNZP 0E (BF7A)
BF6D C3  DECB
BF6E 29  JRNZM 08 (BF67)
BF70 48  INCK
BF71 2B  JRNCM 0D (BF65)
BF73 C8  INCL
BF74 2B  JRNCM 10 (BF65)
BF76 FA  CAL 1ACF
BF78 2D  JRM    5B (BF1E)
BF7A 95  LP15
BF7B 61  ORIM   20
BF7D F1  CAL 11E5
BF7F 37  RTN
```

teste si 0 fin de buffer
si oui LCD OFF: RTN

sauve X dans [1C], [1D]
remplit ram vidéo avec [20] à [37]
LCD ON

scrutation du clavier
teste si BRK
teste si ENTER
teste si SPC
si oui arrêt du décalage

teste si on est en WAIT n
si oui Disp n

~~sauve A~~ ; code touche enfoncée
charge les paramètres de temporisation
fiixe dans K et L
A ←— F1
pour ▶ décalage pas à pas
récupère code touche dans A
teste si ▶
si oui va dans boucle tempo
si aucune touche ; boucle
teste si ▶
si oui décalage rapide

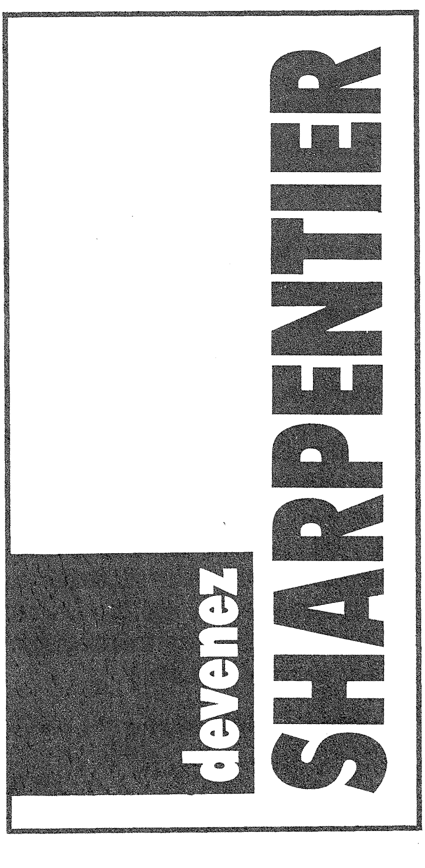
A ←— 0
B ←— 0
charge 0 dans K et L
pointeur paramètres WAIT
chargement des data de (DP)
dans B, A pour Disp n
[L,K] – [B,A] ——▸ [L,K]
si WAIT 0 ; LCD OFF, RTN

Boucle de temporisation

récupère X pour prochain
décalage ; boucle ——▸ fin buffer
flag de BRK
fait afficher un message
LCD OFF
revient au basic

```
DISP N
(C) JFV

BF00 10C6 D7D6 0828 7911
BF08 E757 6719 2A0F F1E0
BF10 FF44 6707 3865 6700
BF18 3864 2D0B F19A 24A0
BF20 0017 1857 6700 3856
BF28 F200 7840 0CF1 E0FF
BF30 4467 0738 4667 0038
BF38 4567 1138 0810 C6D8
BF40 D604 380F 3488 03FF
BF48 56F1 EA5B 670E 3816
BF50 2D22 670E 3821 2303
BF58 0088 F1EA 10C6 E582
BF60 1A88 1538 1903 194E
BF68 5F6B 0828 0EC3 2908
BF70 482B 0DC8 2B10 FACF
BF78 2D5B 9561 20F1 E537
```



METTEZ VOS BOUCLES AU PAS

Suite à l'article sur la boucle FOR/NEXT du numéro 8, voci deux petits programmes qui exploitent des possibilités entrevues lors de cet article, et inexploités dans le PC 1251 ; censure ou bogue ?
Le premier programme est en LM ESR-H, rapide, concis (25 octets), il comporte néanmoins l'inconvénient d'être plus hermétique à la compréhension de tous. de plus, il faut rester prudent quant à son emploi – risque de plantage si il est effacé – contrairement à son homologue en basic.

Mais quelle est donc cette amélioration ? Suspense.
Allez, je vous le dis : il permet d'utiliser comme incrément (STEP) n'importe quel nombre, décimal ou un quelconque réel.

J'exposerai simplement ici son utilisation, le principe du pgm étant assez simple. Vous pouvez présenter vos boucles comme indiqué dans les fig. 1, 2 ou comme sur la fig. 1 bis. Mais il est important de respecter deux contraintes :

- L'incrément doit être stocké dans la var A, donc ne surtout pas utiliser l'instruction STEP.
- Le CALL ou GOSUB doit être la première instruction dans la boucle, et la suite de ladite boucle doit figurer sur la même ligne que CALL ou GOSUB.

De plus il est utile de savoir que le pgm LM est entièrement relogeable et le basic MERGEable à volonté.
La boucle est alors bouclée.

V. CREUX

```
(C) VALERY CREUX

3F00 10C6 98A0 0007 1811
3F08 F357 750D 84DB 02C7
BF10 50DB 04A0 1984 7007
BF18 0457 7407 522A 0504
3F20 5742 5237 0000 0000
```

```
996:"@"B= PEEK &C6F3-12+
    &C700:C=&C698
997:POKE B, PEEK C:B=B+1
    :C=C+1: IF C<&C6A0
    GOTO 997
998:C= PEEK B:C=C+5: IF
    C>255 LET C=C-255:
    POKE B+1, PEEK (B+1)
    +1
999:POKE B,C: RETURN
```

```
10:A=.1: FOR I=0 TO 2:
   CALL &BF00: PRINT I:
   NEXT I
20:END
```

fig.1

```
10:A=.1: FOR I=0 TO 2
20:CALL &BF00: REM ETC
```

fig.1 bis

```
10:A=.1: FOR I=0 TO 2:
   GOSUB "@": PRINT I:
   NEXT I
20:END
```

fig.2

```
(C) VALERY CREUX
BF00 10  LIDP  C698
BF03 A0  LP20
BF04 00  LII   07
BF06 18  MVWD
BF07 11  LIDL  F3
BF09 57  LDD
BF0A 75  SBIA  0D
BF0C 84  LP04
BF0D DB  EXAM
BF0E 02  LIA   C7
BF10 50  INCP
BF11 DB  EXAM
BF12 04  IX
BF13 A0  LP20
BF14 19  EXWD
BF15 84  LP04
BF16 70  ADIM  07
BF18 04  IX
BF19 57  LDD
BF1A 74  ADIA  07
BF1C 52  STD
BF1D 2A  JRNCP 05 (BF23)
BF1F 04  IX
BF20 57  LDD
BF21 42  INCA
BF22 52  STD
BF23 37  RTN
```

A → ram interne

calcul poids faible de l'adr de début de l'incrément dans la pile FOR NEXT

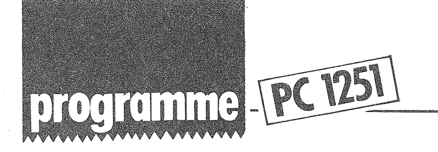
calcul poids fort
adr → X → DP

échange Ram interne avec ancien incrément

mise à jour de X

décalage pointeur de début de boucle

retenue



JEU DE SIM A 8 POINTS

En fin un jeu de réflexion en LM ESR-H sur PC 1251/45.
Ce programme hybride comprend une partie principale en basic, et un algorithme entièrement en LM pour la vitesse d'éxécution.
Le programme est limité à 8 points pour ce jeu de SIM. En effet, au départ il y a 28 coups jouables possibles, et le PC met moins de 2 sec. pour donner les 3 premiers, alors qu'en basic il faudrait 5 mn.

En simplifiant la théorie, chaque coup licite est évalué en comptabilisant — le coup étant joué — ceux qui restent possible au PC moins ceux qui resterait possibles à son adversaire. A chaque coup joué par la machine, la valeur V est affichée. Si elle est positive le PC gagne, si elle est négative il perd.
Si n coups arivent à égalité, l'ordinateur choisit au hasard parmi les meilleurs coups (ayant la meilleure valeur).

Pour jouer, il suffit de répondre aux questiions posées, les coups étant saisis par un INKEY$ amélioré façon GET. Si on ne sait que jouer, taper 0 et la machine suggérera un coup jouable... on annoncera la défaite.
Pour adapter ce pgm au PC 1245 (ou le reloger), se servir du relogeur basic fourni, en faisant attention à garder les mêmes poids faibles aux adresses. Relogez en XX20. Evidemment, il faut modifier en conséquence les adresses dans le basic. (Voir relogeur dans n° 9).

G. NICOLAS

```
  5:"* SIM A 8 POINTS *"
  6:"(C) GILLES NICOLAS"
 10:CALL &BE20
 20:INPUT "TU COMMENCES
     ?(O/N)";Z$
 30:IF Z$="N" THEN 90
 40:I$="":J$="": WAIT 0
 42:PRINT "A TOI DE JOUE
     R ": CALL &1D23:I$=
     INKEY$ : IF I$=""
     CALL &11E5: GOTO 42
 45:IF I$="0" GOTO 160
 50:PRINT "A TOI DE JOUE
     R ";I$: CALL &1D23:J
     $= INKEY$ : CALL &11
     E5
 55:I= VAL I$:J= VAL J$
 60:O= PEEK (&C000+I):H=
     PEEK (&C008+I):K=
     PEEK (&C008+J):P=
     PEEK (&C010+J):Q=
     PEEK (&C010+I)
 70:IF ((O OR H) AND P)
     OR (H AND K) BEEP 1:
     PAUSE "COUP ILLEGAL"
     : GOTO 40
 80:POKE &C008+I,H+P:
     POKE &C008+J,K+Q
 90:CALL &BED3:M= PEEK &
     C01A
100:IF M=0 BEEP 2: PRINT
     "TU AS GAGNE !BRAVO
     !": END
110:RANDOM :M= RND M:
     WAIT 128
120:I= PEEK (&C019+2M)+1
     :J= PEEK (&C01A+2M)+
     1:V= PEEK &C019-127
130:POKE &BFFE,I-1,J-1:
     CALL &BF47
140:PRINT "JE JOUE ";
     STR$ I;" -- "; STR$
     J;" (V="; STR$ V;")"
150:GOTO 40
160:WAIT : CALL &BF58:
     IF PEEK &C019=255
     PAUSE "TU PEUX JOUER
      "; STR$ ( PEEK &C01
     A+1);" - "; STR$ (
     PEEK &C01B+1): GOTO
     40
170:BEEP 3: PRINT "TU AS
      PERDU !": END
```

```
SIM A 8 POINTS
(C) GILLES NICOLAS

BE20 0010 0200 10C0 001F
BE28 8702 C0DB 5102 10DB
BE30 0207 3402 0126 575A
BE38 2F04 3785 02C0 DB00
BE40 0084 1330 08A0 0455
BE48 8470 07A0 2447 1331
BE50 8408 5170 10A0 2446
BE58 3784 1330 08A0 0455
BE60 8413 3108 A024 4637
BE68 B002 0870 08B1 0208
BE70 7008 78BE 5937 8413
BE78 0808 A304 5584 700F
BE80 A404 5513 0984 08A4
BE88 2447 8470 0F24 A347
BE90 1308 8408 04A3 1913
BE98 0984 0804 A419 37A2
BEA0 027F DB90 0208 DB90
BEA8 7101 2802 3713 1091
BEB0 0891 7101 3B0E B013
BEB8 1008 0878 BE3B 290E
BEC0 78BE 5928 04A2 7001
BEC8 78BE 6828 04A2 7101
BED0 2D20 3787 02C0 DB86
BED8 0218 DB02 0026 0200
BEE0 2688 0208 DB49 2802
BEE8 3713 0889 08C9 3B0A
BEF0 3013 0808 0878 BE3B
BEF8 290C 78BE 5929 1178
BF00 BE76 78BE 9F13 0884
BF08 08A3 0419 1309 8408
BF10 A404 1987 02C0 DBA2
BF18 10C0 1957 C77F BEED
BF20 57C7 3814 8602 18DB
BF28 A206 5302 0126 8859
BF30 2689 5926 79BE ED88
BF38 5926 8959 2610 C01A
BF40 5774 0152 79BE ED85
BF48 02C0 DB00 0110 BFFE
BF50 8818 0000 78BE 7637
BF58 9002 08DB 9071 0128
BF60 0237 1310 9108 9171
BF68 013B 0EB0 1310 0808
BF70 78BE 3B29 0E78 BE68
BF78 2913 8502 C0DB 8402
BF80 18DB 0402 FF52 9004
BF88 5391 0453 3700 0000
```

```
RELOGEUR SIM

 10:INPUT "@DR:";A
 20:FOR I=1 TO 18: READ
     B,C:D= INT ((A+C)/25
     6): POKE A+B,D,A+C-2
     56D: NEXT I
 30:FOR I=1 TO 6: READ B
     :D= INT ((A+&200)/25
     6): POKE A+B,D: NEXT
     I
 40:DATA 5,480,83,57,156
     ,27,161,57,169,72,21
     4,27,219,57,224,86,2
     27,127,249,505,254
 50:DATA 205,277,205,286
     ,506,293,205,302,478
     ,309,86,337,27,342,7
     2
 60:DATA 10,29,181,245,2
     97,348
```

PROGRAMME DE NAVIGATION ASTRONOMIQUE

LE « POINT » ASTRONOMIQUE EN NAVIGATION

Un relevé sextant permet, toutes corrections faites, de mesurer la hauteur d'un astre au-dessus de l'horizon (tale). Nous appellerons cette hauteur S. Sachant que la terre est ronde, il est évident qu'à un instant donné il n'y a pas qu'un seul point sur la terre d'où l'on voit cet astre à la hauteur S en-dessus de l'horizon. En effet (cf. fig. 1) en tous les lieux situés sur un cercle justement appelé cercle d'égale hauteur, on mesurerait la même hauteur S. Ce cercle admet pour axe la droite qui joint le centre de la terre à celui de l'astre.

Le « point » astronomique situe donc l'observateur sur un cercle. ce cercle est défini par son centre et son rayon (cf. fig. 2).
• le centre est repéré par deux angles qui sont l'angle horaire (AHo en français, GHA en anglais) et la déclinaison (déc.).
• le rayon est égal à l'angle complémentaire du sextant (90-S).

Figure 1 : Cercle d'égale hauteur. En tous les points du cercle un observateur mesurerait la même valeur S pour la hauteur de l'astre au-dessus de l'horizon.

Figure 2 : Représentation plane (sur une carte) du cercle d'égale hauteur; Un « POINT » astronimique situe l'observateur sur un CERCLE.

TRACE SUR LA CARTE

La méthode classique (Intercept et Azimuth) consiste à estimer sa propre position (latitude et longitudes estimées) et à tracer une portion du cercle d'égale hauteur par rapport à ce point (cf. fig. 3).
Le cercle est localement assimilé à une droite (la fameuse droite de hauteur). Cette construction nécessite de reporter sur la carte :
• le point estimé (lattitude et longitude)
• un angle (l'Azimuth A qui est la direction de l'astre par rapport au nord).
• une distance (l'Intercept I qui représente la différence entre la valeur corrigée du sextant et la hauteur calculée de l'astre)
• et demande de tracer au point ainsi trouvé la perpendiculaire à l'Azimuth (ne pas oublier). Cette possibilité est incluse dans le programme, mais nous proposons une méthode beaucoup plus simple : pour une latitude donnée, la calculatrice donne les deux longitudes possibles (cf. fig. 4). L'estime permet généralement de trier entre les deux longitudes proposées, mais dans les cas extrêmes (astre proche du sud local), les deux longitudes peuvent être portées sur la carte.
A l'aide d'une deuxième latitude (demandée automatiquement par la calculatrice), on obtient deux autres longitudes. Pour obtenir la droite de hauteur, il suffit de joindre les deux points correspondants (cf. fig. 4). Cette construction ne demande aucun report d'angle ou de distance, une simple règle non graduée suffit.

Dans tous les cas de figure, les calculs nécessitent la connaissance préalable :
• de la déclinaison et de l'angle horaire de l'astre
• des corrections à apporter à la mesure sextant

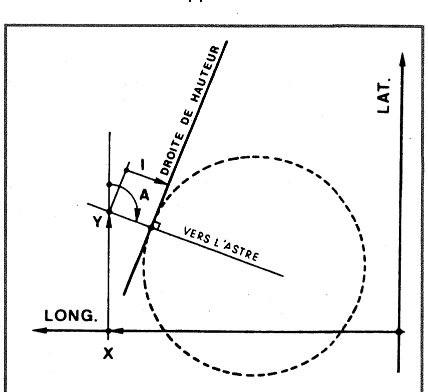


Figure 3 : Construction de la droite de hauteur. Pour un point estimé de longitude X et de latitude Y, le calcul donne :
• l'Azimuth A (direction de l'astre par rapport au nord,
• l'Intercept I (porté vers l'astre si I est positif, en sens inverse si I est négatif).
Cette construction situe l'observateur sur une droite de hauteur ; cette droite constitue généralement une approximation suffisamment bonne du cercle.

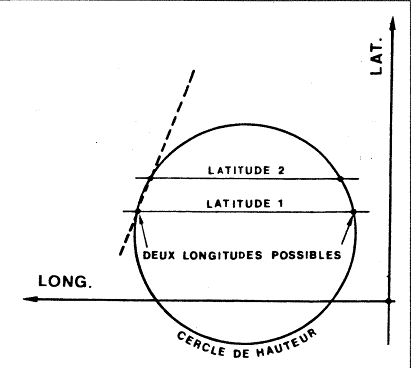


Figure 4 : Tracé du cercle d'égale hauteur. A chaque latitude correspondent deux longitudes. L'estime et l'azimuth permettent de déterminer la longitude adéquate sauf dans des cas extrêmes où justement il vaut mieux représenter le cercle que la droite de hauteur qui devient alors une mauvaise approximation.
Le programme demande deux latitudes. Si on voulait tracer la totalité du cercle il suffirait de repasser le nombre de fois voulu sur la ligne 100.

LE SOLEIL

Pour le soleil qui est de loin l'astre le plus utilisé, du moins en navigation de plaisance, le calcul du point est entièrement programmé. Les calculs ne sont évidemment jamais parfaits et c'est pourquoi nous présentons sur la figure 5 l'écart entre les valeurs calculées et les valeurs des

éphémérides, pour la déclinaison et l'angle horaire du soleil, de 1978 à 1984. La déclinaison est donnée avec une précision de l'ordre du dixième de minute d'angle soit deux secondes de temps.

Ces écarts sont dus aux perturbations par les planètes, non prises en compte dans le programme. Globalement l'erreur sur le calcul du point ne dépasse pas un demi-mille marin. Si une mise à jour s'avérait nécessaire pour l'angle horaire, il suffirait de modifier très légèrement une seule valeur (voir MISE A JOUR).

LUNE, PLANETES ET ETOILES

Les éphémérides sont nécessaires pour la lune, les planètes et les étoiles (mais ne sont-elles pas toujours présentes sur un bateau ?). Les entrées demandent un effort réduit au minimum et le programme effectue les interpolations nécessaires, ce qui n'est pas aussi évident qu'il y paraît, afin d'obtenir la déclinaison et l'angle horaire de l'astre.

REFRACTION, DEPRESSION, ETC.

Toutes les corrections à effectuer (dépression de l'horizon, réfraction, demi-diamètre et parallaxe du soleil ou de la lune) sont incluses dans le programme pour toutes les hauteurs d'astres observables. Les valeurs numériques, notamment celles de la réfraction ont été prises dans l'annuaire du bureau des longitudes. Les deux expressions analytiques que nous proposons pour la réfraction atmosphérique se raccordent parfaitement et s'écartent de la correction idéale de moins de 0.05' d'angle dans la gamme de hauteur 0 à 90 degrés.

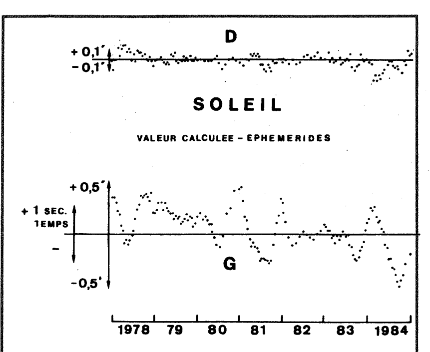


Figure 5 : Ecarts entre les valeurs calculées par le programme et les valeurs fournies par les éphémérides pour la déclinaison D et l'angle horaire G du soleil.

Il pourrra sembler curieux de donner des corrections de réfraction pour des visées proches de l'horizon, Nous voudrions cependant faire deux remarques :

- Tout d'abord, pour quelle limite de hauteur au-dessus de l'horizon fait-il s'arrêter ?
- Ensuite même si les conditions réelles de réfraction s'écartent quelque peu des conditions idéales, il n'en reste pas moins qu'un relevé obtenu pour des hauteurs faibles est toujours mieux que rien (de plus, quoi de plus agréable que de relever sans sextant un lever ou un coucher d'astre).

Dans le même ordre d'idées, les corrections de pression et de température sont généralement superflues. Mais il faut savoir qu'aux alentours de dix degrés de hauteur elles peuvent représenter 0.5' d'erreur. De toute façon en l'absence de réponse du navigateur, la calculatrice prendra des conditions moyennes (si vous naviguez sur le lac Titicaca entrez quand même la pression locale !).

AUTRES UTILISATIONS DU PROGRAMME

CALCUL D'UNE ROUTE

On peut très facilement calculer la distance et l'azimuth pour une route. Pour cela faire (en mode calcul (RUN) et surtout pas PROGRAMMATION) :

S = 90 ENTER
D = DEG (latitude de la destination en °.'.") Enter
G = DEG (latitude de la destination en °.'.") Enter
RUN100 Enter

Le programme demande alors la latitude et la longitude qui sont celles du point de départ. Il donne ensuite les résultats suivants :

A = CAP DEPART Route orthodromique, en degrés décimaux)
I = DISTANCE (en milles marins)

CHERCHER LES ETOILES ET VERIFIER LES BOUSSOLES

Pour fonctionner en chercheur d'étoile, il suffit d'exécuter le programme, sans répondre aux questions relatives au sextant, jusqu'à la ligne 105 (Azimuth et Intercept). L'azimuth étant connu, pour obtenir la hauteur, il suffit d'arrêter le programme et de demander la variable Z qui représente la hauteur (non corrigée) de l'astre en degrés décimaux :

BREAK
Z ENTER

En sens inverse, si l'astre est identifié et si la position du bateau est bien connue, l'azimuth des astres proches de l'horizon peut servir à calibrer de façon très précise les compas de navigation.

PASSAGE D'UN ASTRE AU SUD LOCAL

Pour obtenir l'heure de passage au sud local, exécuter le programme en entrant comme heure une valeur approximative de l'heure de passage sans répondre aux différentes questions relatives au sextant.

Après l'affichage de l'azimuth et de l'intercept, exécuter la séquence suivante (en restant bien sur en mode calcul ou RUN) :

BREAK
H = (H + W - U / 15) / 24 ENTER
H = DMS (24* (H - INT H)) ENTER

Le résultat affiché est alors l'heure de passage en heures, minutes, secondes. Pour une meilleure précision et notamment pour la lune, refaire un passage en faisant RUN 10 puis ENTER. A la question heure, répondre H (lettre H).

MISE A JOUR DES EPHEMERIDES DU SOLEIL

Pour recaler l'anglé horaire du soleil, il suffit de faire dérouler le programme (sans répondre aux questions relatives au sextant) pour la date à laquelle on veut coller le mieux avec les éphémérides nautiques. Lorsque la calculatrice demande la latitude, faire :

BREAK
G
ENTER

On obtient à l'affichage l'angle horaire G du soleil en degrés décimaux. Il suffit alors de la comparer à la valeur des éphémérides (que l'on aura pris soin de transformer en degrés décimaux : attention aux minutes décimales des

éphémérides nautiques). Si l'écart est trop important pour l'utilisation prévue il faut alors modifier la valeur de la variable Q ligne 6 :

- l'augmenter de la différence si G est plus petit
- la diminuer de la différence si G est plus grand.

La déclinaison est beaucoup moins sensible aux perturbations. S'il fallait vraiment la retoucher opérer comme précédemment avec la variable Y (BREAK Y ENTER) que l'on comparera à la valeur donnée par les éphémérides de la déclinaison du soleil pour le SOLSTICE le plus proche.

REMARQUES

Le calcul de la position du soleil est « valable » de 1901 à 2099. A partir de l'année 2000 (!), il faut toutefois remplacer à la ligne 8 le morceau d'instruction :

N - 81

par N + 19

Il faudrait aussi pour vraiment bien faire modifier certaines grandeurs telles que l'excentricité, tenir compte de la rotation du périgée..., mais gageons que d'ici là des programmes et des machines bien plus astucieux seront à notre disposition.

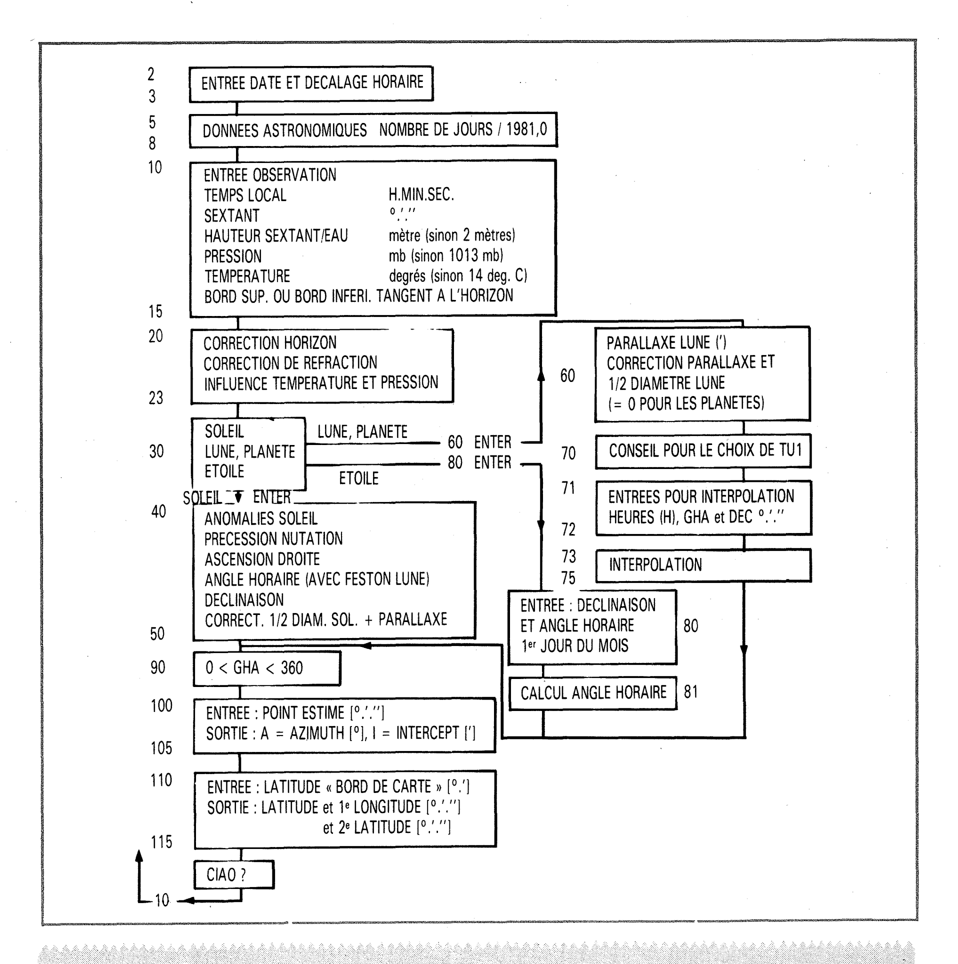
Afin d'éviter les confusions entre la lettre O et le zéro, il suffit de savoir que la variable O (lettre O) n'intervient qu'aux lignes 6 et 40.

Sur la ligne 46, 5E-6 signifie : 5/1 000 000. Ce terme représente le feston dû à la lune. Pour être exact il faudrait soustraire la valeur de l'expression pour T = P (origine des rotations) mais dans notre cas cette correction est inutile (0.003').

EN GUISE DE CONCLUSION

Ce programme est optimisé pour le soleil et pour une utilisation sur calculatrice SHARP PC 1211/1212 (1424 pas de programme). Cette calculatrice ne demande ni de fermer les parenthèses en fin de ligne, ni de mettre le signe de la multiplication entre les variables, ce qui fait gagner quelques 30 à 40 pas. Sur les 1424 pas, il en reste encore une trentaine libres, ce qui suffit pour entrer la courbe de correction propre du sextant (ligne 15).

Pour ceux qui ne s'intéressent qu'au soleil et au seul calcul de l'azimuth et de l'intercept, nous tenons à leur disposition un programme compact de 750 pas de programme. De toute manière n'oubliez pas d'emporter les éphémérides nautiques avec vous !




```
1    "N"
2    INPUT "JOUR=" ; J ,"MOIS=" ; M , "ANNEE NN=" ; N
3    INPUT "TEMPS LOCAL - TU =" ; W
5    C = 23.4420  : E = .016710  : L = 77.383   : B = 365.256363
6    O = 2.088548 : P = 78.71081 : Q = 074.0310 : R = 360 :DEGREE
8    I = J + 31M - 32 -(M>2) * (3 -(INT (N/4) = N/4 ))
       - (M>4) - (M>6) - (M>9) - (M>11) + INT (B*(N-81
10   INPUT "TEMPS H.MNS =" ; H ,"SEXTANT D.MNS ="; S
11   Z = 2.0 : X = 1013 : Y = 14 : INPUT "HAUT/EAU" ; Z ,
     "P(MB)=" ; X,"T(DC)=" ; Y
12   F = -1 : INPUT "BORD INF = ENTER, SUP = 1 " ; F
15   H = DEG H - W : T = I + H/24 : S = DEG S
20   S = S - √(Z/1136
21   IF S > 11 LET Z = 1 / TAN S / 61 : GOTO 23
22   Z = .61 / (1 + .4S + SS/33 * (1-S/22
23   S = S - Z X/1013 / (1 + Y/258
30   INPUT "SOL=ENT,LU=PL=60,ET=80 " ; Z : GOTO Z
40   FOR Z = 1 TO 4 : U = (T-O)*R/B + ER/2π * SIN U : NEXTZ
41   V = 2 * ATN (TAN (U/2) *  √((1 + E)/(1 - E
43   Z = T - P : K = 152 - .053T : X = .014 * (Z/B - (SIN K - .53)/3:
     Y = C + .0026 * (COS K -T/20B
45   K = V - L + X : A = ATN (TAN K * COS Y) + R/2 * (ABS K > 90
46   G = H/24-P + Z/B + (Q - A + X*COS Y) / R
       - 5E-6 * (SIN (R * (T-5) / 29.53
48   D = ASN ( SIN Y * SIN K
50   S = S + .267 * (1 + E * COS V) * (COS S / 109 - F): GOTO 90
60   Z = 0 : INPUT "PRLX=" ; Z : S = S + Z/60 * (COS S - .272F
70   PRINT USING "###.#"; "TU1 < " ; H
71   INPUT "TU1=" ; K, "GHA1=" ; G, "D1=" ; D
72   INPUT "TU2-TU1=" ; Z, "GHA2= ; U, "D2=" ; V
73   G = DEG G / R : U = DEG U / R : D = DEG D : V = DEG V
74   U = U + (Z<24)*(U<G) + (Z=24)*(1+(U<.1)*(G>.9)-(U>.9)*(G<.1
75   Z = (H-K)/Z : G = G + Z*(U-G : D = D + Z*(V-D : GOTO 90
80   INPUT "J=1 H=OTU D=" ; D, "GHA=" ; G : D=DEG D : G=DEG G
81   G = G/R + (1 + 1/B) * (J - 1 + H/24
90   G = R * (G - INT G
100  INPUT "LAT D.MNS (N+,S-)=" ; Y, "LONG (O+,E-)=" ; X
101  V = DEG Y : U = G - DEG X
102  Z = ASN ( COS V * COS D * COS U + SIN V * SIN D
103  A = (SIN V * COS D * COS U - COS V * SIN D) / COS Z
104  A = SGN SIN U * ACS A + R/2 : K = 60*(S - Z
105  PRINT USING "####.#" ; " A =" ; A; "    I=" ; K
```

```
110  FOR F = 1 TO 2 : INPUT "LAT=" ; Y : V = DEG Y
111  U = (SIN S - SIN D * SIN V) / COS D / COS V
112  IF ABS U > 1 BEEP 5 : GOTO 110
113  X = G - ACS U : Z = 2G - X
114  X = DMS (X - R*(2X>R : Z = DMS (Z - R*(2Z>R
115  PRINT USING "###.# " ; Y ; USING"####.####"; X ; Z : NEXT F
200  PRINT "CIAO ?" : GOTO 10
```

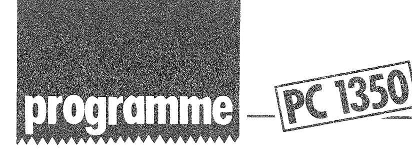
Exemple

Localité : ANTILLES - Sainte-Lucie - Deux-Pitons
RUN
JOUR = 31
MOIS = 12
ANNEE NN = 84
TEMPS LOCAL – TU = –4

TEMPS H.MNS = 16.24.16
SEXTANT D.MNS = 17.02
HAUTEUR / EAU = 2
P (MB) = 1000
T (DC) = 28
ENTER
ENTER

LAT DMS (N+, S–) = 13.50
LONG (0+, E–) = 61.04
A = 240.1 I = 0.5
ENTER
LAT = 13.50
13.50 61.04 40 –170.37 00
ENTER
LAT = 14.00
14.00 61.10 35 –170.42 56
ENTER
CIAO ?

	ASTRONOMIE SOLEIL (1–< 50)	SEXTANT (10,15,20...23,50,60) (104,111)	INTERPOLATION 70–< 75) (80–< 81)	NAVIGATION (100–< 105) (110–< 115)
A	Ascension droite			Azimuth
B	Année sidérale			
C	Inclinaison écliptique moyenne 1981,0			
D	Déclinaison		Déclinaison (TU1)	Déclinaison
E	Excentricité 1990			
F		Bord sup. ou bord inf.		
G	GHA = AHo = angle horaire/ Greenwich		GHA (TU1)	GHA
H	Heure d'observation locale (10), TU (15)			
I	Nb jours entiers / 1981,0			
J	Jour du mois			
K	Angle de nutation (43) Longitude héliocentrique (45)		TU1	Intercept
L	Longitude périgée 1981			
M	Mois			
N	Année (deux derniers chif.)			
O	Temps périgée 1981			
P	Temps printemps 1981			
Q	GHA soleil pour T = P			
R	360 degrés/tour		360	360
S		Valeur lue –> Valeurs corrigées		Sextant corrigé
T	Temps d'observation / 1981,0			
U	Anomalie excentrique		GHA (TU+)	Angle horaire local
V	Anomalie vraie		Déclinaison (TU2)	Latitude
W	Temps local moins temps universel			
X	Précession + nutation (nutation = 0 pour T = P –> 53)	Pression (mb)		Longitude estimée (100) 1ere long. possible (115)
Y	Inclinaison écliptique vraie	Température (degrés C)		Latitude estimée (100) Lat. « bord de carte » (110)
Z	Temps/printemps 81	Hauteur/eau (m)	TU2-TU1, (T-TU1) / (TU2-TU1)	Hauteur astre (104) 2e long. possible (115)



JACK POT

DING DINGING DING...

Il ne manque plus que le bruit des pièces pour que l'on se croit devant une de ces célèbres machines à sous.

Pour jouer, il faut d'abord miser une somme allant de 1 à 9 francs puis, pour arrêter les rouleaux, on doit presser les touches R, T, Y (chaque lettre correspondant au rouleau qui se trouve au dessus d'elle). On peut si on le désire laisser la machine s'arrêter toute seule.

SIX SYMBOLES SONT PROPOSES :

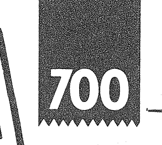
Une cloche, une pomme, un citron, deux cerises, trois bananes, et une barre.

On gagne en multipliant sa mise par un coëfficient qui correspond aux combinaisons suivantes :

- 1000 fois la mise pour
une barre (R), une cloche (T) et une cloche (Y)
- 100 fois la mise pour
une cloche (R), une cloche (T) et n'importe quoi (Y)
- 30 fois la mise pour
trois bananes (R), trois bananes (T) et n'importe quoi (Y)
- 10 fois la mise pour
une cloche (R), et n'importe quoi (T) (Y)
- 5 fois la mise pour
une barre (R), et n'importe quoi (T) (Y)
- 2 fois la mise pour
trois bananes (R), et n'importe quoi (T) (Y)
- 1 fois la mise pour
une pomme (R), et n'importe quoi (T) (Y)

```
5:Y= INT (X/24):X=(X-Y
  *24)*6+6:Y=(Y+1)*8-1
  : GCURSOR (X,Y):
  FOR I=1 TO LEN A$:S=
  ASC MID$ (A$,I,1)*5+
  &7FEE
6:WAIT 0: FOR J=S TO S
  -4 STEP -1: GPRINT
  PEEK J;: NEXT J:
  GPRINT 0;: NEXT I:
  RETURN
5:Y= INT (X/24)*8:X=(X
  -Y*3)*6+6: GCURSOR (
  X,Y+7): FOR I=1 TO
  LEN A$:S= ASC MID$ (
  A$,I,1)*5+&7FEE
6:WAIT 0: FOR J=S TO S
  -4 STEP -1: GPRINT
  PEEK J;: NEXT J:
  GPRINT 0;: NEXT I:
  RETURN
```

```
4010:REM ++++++++++++
4020:REM + JACK POT +
4030:REM ++++++++++++
4040:CLEAR : DIM D$(2)*
     20,P$(5)*16:
     RANDOM
4050:CLS : WAIT 0:
     PRINT "----J-A-C-K
     ---P-O-T----"
4060:P$(0)="20787C3E3E7
     C7820"
4070:P$(1)="3C3C3C3C3C3
     C3C3C"
4080:P$(2)="78451F674D1
     42040"
4090:P$(3)="387C7D7D7F7
     C7C38"
4100:P$(4)="183C7E7E7E7
     E3C18"
4110:P$(5)="70707008047
     37C70"
4120:D$(0)="40514254315
     423523435"
4130:D$(1)="45054515452
     541424245"
4140:D$(2)="31534504534
     134534534"
4150:C=30:X= RND 20:Y=
     RND 20:Z= RND 20
4160:LINE (45,9)-(88,22
     ),S,B
4170:FOR I=0 TO 2:
     LINE (46+I*14,10)-
     (59+I*14,21),S,B:
     NEXT I: CURSOR 19,
     2: PRINT "*Mise"
4180:CURSOR 0,3: PRINT
     "Total:"; USING "#
     #####";C
4190:CURSOR 13,3:
     USING : PRINT "Mis
     e(1-9)?"
4200:A= VAL INKEY$ :
     IF A=0 OR A>C
     GOTO 4200
4210:BEEP 1:C=C-A:
     CURSOR 6,3: PRINT
     USING "######";C
4220:USING : CURSOR 23,
     3: PRINT STR$ A;:
     CURSOR 14,2:
     PRINT "     "
4230:I=13:J=14:K=15
4240:IF I GCURSOR (49,1
     9): GPRINT P$(
     VAL MID$ (D$(0),X,
     1)):I=I-1:X=X+1:
     IF X=21 LET X=1
4250:IF J GCURSOR (63,1
     9): GPRINT P$(
     VAL MID$ (D$(1),Y,
     1)):J=J-1:Y=Y+1:
     IF Y=21 LET Y=1
4260:IF K GCURSOR (77,1
     9): GPRINT P$(
     VAL MID$ (D$(2),Z,
     1)):K=K-1:Z=Z+1:
     IF Z=21 LET Z=1
4270:IF I>1 AND INKEY$
     ="R" LET I=1:
     BEEP 1
4280:IF J>1 AND INKEY$
     ="T" LET J=1:
     BEEP 1
4290:IF K>1 AND INKEY$
     ="Y" LET K=1:
     BEEP 1
4300:IF I OR J OR K
     GOTO 4240
4310:I=X-1: IF I=0 LET
     I=20
4320:J=Y-1: IF J=0 LET
     J=20
4330:K=Z-1: IF K=0 LET
     K=20
4335:I= VAL MID$ (D$(0)
     ,I,1):J= VAL MID$
     (D$(1),J,1):K=
     VAL MID$ (D$(2),K,
     1)
4340:Q=0: IF I=3 LET Q=
     1
4350:IF I=2 LET Q=2
4360:IF I=1 LET Q=5
4370:IF I=0 LET Q=10
4380:IF I=2 AND I=J
     LET Q=30
4390:IF I=1 AND I=J
     LET Q=50
4400:IF I=0 AND I=J
     LET Q=100
4410:IF I=1 AND I=J
     AND I=K LET Q=500
4420:IF I=0 AND I=J
     AND I=K LET Q=1000
4430:CURSOR 14,2:
     PRINT USING "#####
     ";Q;: USING :
     CURSOR 23,3:
     PRINT " ";
4440:C=C+A*Q: BEEP 1:
     IF C>0 GOTO 4180
4450:WAIT : CURSOR 13,3
     : PRINT "*GAME OVE
     R*";: END
```



K BASIC V.5

Le K-BASIC est un interpréteur destiné à la programmation de l'ordinateur SHARP MZ-700. Il a été obtenu par modification et extension du S-BASIC livré avec l'ordinateur et, par suite, sera en grande partie compatible avec celui-ci.

Le K-BASIC se présente sous deux aspects :
• C'est tout d'abord un interpréteur Basic « standard », possédant toutes les instructions du S-BASIC avec, de plus :
ELSE pour les tests,
INKEY et KEY complétant l'ordre GET,
MAX, MIN, OR, AND, XOR, DIV MODULO, ASN, ACS pour les calculs,
GRAPH, DRAW, POINT sur le semi-graphisme sur l'écran,
SWAP, NULL, COPY pour manipuler des tableaux (à moins de 128 dimensions...),
WAIT et PWAIT pour ralentir l'exécution des programmes,
INV§, IN§, STRING§, INSTR pour manipuler des chaînes de caractères,
VPOS et HPOS pour trouver la position du curseur,
DISP pour afficher, en programme, une ligne de programme,
EVAL permettant d'appliquer l'évaluateur de formules au contenu d'une chaîne de caractères.
• C'est surtout un interpréteur destiné à écrire des programmes selon les règles de la « programmation structurée ». Dans ce type de programmation, bien connue des utilisateurs de langages de haut niveau (comme PASCAL), le programme sera construit de façon modulaire, chaque module pouvant être mis au point séparément. Avec un peu d'habitude, on pourra alors écrire de longs programmes, faciles à mettre au point, à modifier et dont la structure (et par suite, le mode de fonctionnement) se reconnaît au premier coup d'œil sur un listing. Surtout, on ne trouvera plus ces nombreux GOTO renvoyant d'un bout à l'autre du programme et le rendant incompréhensible à tout autre qu'à l'auteur, et même à celui-ci, quelques mois plus tard !
Que le lecteur se rassure ! Le K-BASIC permettant les deux types de programmation, il pourra toujours écrire un programme en suivant les « mauvaises » habitudes dues à l'utilisation des Basics standards, puis le modifier, l'améliorer pour lui donner une forme plus structurée. Ainsi, petit à petit, il lui sera possible de s'habituer à la programmation structurée et de se rendre compte de la grande puissance de cette méthode de programmation.
Ce type de programmation est permis par la présence de :
3 types de boucles :
FOR NEXT
REPEAT UNTIL
WHILE WEND
2 types de tests :
IF ... ELSIF ... ELSIF ... ELSE ... ENDIF
CASE...OF...WHEN...WHEN...OTHERWISE... ENDCASE
(Ces deux structures peuvent s'étendre sur un nombre quelconque de lignes)
2 types de sous-programmes
(en plus des GOSUB-RETURN) ;
Les procédures définies par DEF PROC et appelées par PROC
Les fonctions définies par DEF FN et appelées par FN
Et permettant toutes deux le passage de paramètres par valeur et par référence avec, bien entendu, l'utilisation possible de variables locales par LOCAL.
Ces deux structures permettent évidemment la récursivité.
(Remarque : DEF FN existe en S-Basic mais sous une formle beaucoup moins puissante qu'en K-Basic).

Amiens le 1[er] septembre 1984
B.KOKANOSKY

Voici, présenté par son concepteur, Bernard KOKANOSKY, le SUPER BASIC, pour MZ-700.

Pour vous donner une idée plus précise de la puissance et de la qualité de ce nouveau langage, nous nous bornerons à vous préciser le nom et les possibilités des différentes fonctions qui ne possède pas le S-BASIC. K-Basic est dès maintenant disponible au club au prix de 100 F. (cassette + notice + port).

Est-il utile de vous préciser que la conception de ce langage a demandé à son auteur, un travail considérable ainsi qu'une parfaite connaissance de la structure de l'interpréteur BASIC, et du microprocesseur Z80, et que Monsieur Konanowski nous a confié ce logiciel et sa notice (+ de 50 pages) dans un esprit totalement désintéressé, et dans le seul but de fournir un outil puissant capable de tirer un maximum de

GOTO	peut-être donné par une fonction que le basic calculera
GOSUB	aucune limite dans le nombre de sous-programmes imbriqués
DISP n	affiche en programme la ligne N° n
RETURN	possibilité de RETURN n° de ligne (comme S-BASIC)
INREY	attend l'enfoncement d'une touche en faisant clignoter le curseur
KEY	équivalent du GET, mais à répétition
ERROR	structure identique au S. BASIC, mais dans K-BASIC, il y a 59 messages d'erreur en français
ELSE	utilisé dans la structure IF...THEN...ELSE
RESULT	expression : retourne « EXPRESSION » comme résultat d'une fonction FN
BEEP ·	permet la création de sons complexes
GRAPH	sert à placer un pseudo curseur graphique en (x,y)
DRAW	sert à tracer une droite sur l'écran (avec les points « SET »)
SWAP	permute les éléments de 2 tableaux
NULL	effacement sélectif des tableaux (X)
COPY	reproduit le contenu d'un tableau dans un autre
EXC	échange le contenu de 2 variables
WAIT	instruction de temporisation
PWAIT	interrompt légèrement le programme après chaque « PRINT »
LOCAL	
EXIT	permet de sortir d'une ou plusieurs boucles non closes

possibilités du MZ-700, qu'il en soit ici sincèrement remercié.

S. BIZOIRRE

** B. Kokanoski est également l'auteur du SUPER BASIC pour MZ-80K, devenu depuis une référence sur cette machine, en France et à l'étranger.*

SPOKE	permet de colorer des caractères affichés à l'écran
VARTPR	donne l'adresse mémoire ou se trouve une variable recherchée
MAX	A MAX B donne le maximum des 2 variables A et B
DIV	A DIV B = INT (A/B)
INV$	inverse l'ordre des caractères d'une chaîne
VPOS	position verticale du curseur
HPOS	position horizontale du curseur
IN$	permet d'insérer une chaîne de caractères dans une autre chaîne
INSTR	recherche une chaîne dans une autre chaîne
SH	Sinus hyperbolique calculé par (EXT(X) – I/EXP(X))/2
CH	Cosinus hyperbolique calculé par : (EXP(X) + I/EXP(X))/2
TH	Tangente hyperbolique par SH(X)/CH/(X)
ASH	Argument sinus hyperbolique calculé par : LN(X + SQR(I + X x X)) si X est positif ou nul et – ASH(– X) sinon.
ACH	Argument cosinus hyperbolique calculé par : LN(X + SQR(X x – I))
ATH	Argument tangente hyperbolique calculé par : LN((I + X)/(I – X))/2
FACT	Factorielle. Son argument doit être entier positif ou nul.

ASTUCES ASTUCES ASTUCES

LES INSTRUCTIONS CACHEES DU S-BASIC

Ceux d'entre vous qui ont lu le bulletin n° 8, savent déjà que la notice ne mentionne pas toutes les instructions (HEX$, BOOT, CLS, etc.). Mais depuis, de nombreux SHARPENTIERS ont fait des découvertes, et nous ont écrit, nous pouvons donc vous présenter, outre deux nouvelles instructions, des novuelles systaxes pour celle que vous connaissiez déjà.



TRON ET TROFF

Si tous vos programmes tournent du premier coup ces instructions vous seront inutiles. Par contre, si vous êtes de ceux qui passez de nombreuses heures à chercher pourquoi telle boucle ne se termine jamais, ou pourquoi telle variable est remise à zéro alors qu'elle ne devrait pas TRON vous sera très utile. En effet, sit vous entrez en mode TRON, le MZ-700 vous indiquera en permanence quelle ligne il est en train d'exécuter. Pour entrer dans un tel mode, il suffit de taper TRON, pour en sortir TROFF.



LES REM

Vous pouvez, si vous le désirez, les remplacer par des apostrophes : c'est plus rapide à taper et certaines trouvent que c'est plus joli.

Exemple : 10 REM BLABLA
Equivaut à 10' BLABLA.



LES TESTS

Les THEN sont superflus dans la quasi totalité des cas.

Ainsi 100 IF A = 0 END est tout à fait correct.
Cependant IF A = 2 B = 3 déclenche une erreur, il faut écrire IF A = 2 THEN B = 3
ou IF A = 2 LET B = 3



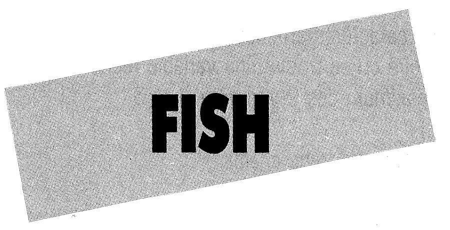
LES RETURNS

Sortir d'un sous-programme pas un GOTO est dangereux (une fois ça va, 16 fois bonjour les dégâts !). Vous pouvez donc écrire :
RETURN N° de ligne qui clôt l'exécution d'un sous-programme tout en permettant de revenir à un endroit indépendant de celui où il a été appelé par GOSUB.



LES PRINT'S

Plus besoin de « ; », vous pouvez écrire en toute tranquillité : 10 PRINT TAB (10) « BONJOUR » SPC (8) « ÇA VA ? ».



Programme pour MZ-80K - MZ-80A - MZ-80B - MZ-700

FISH est un programme transformant votre MZ en une boîte pouvant contenir jusqu'à 255 fiches comportant chacune un maximum de 9 rubriques. Dans ce répertoire, vous pourrez entrer, classer, modifier, trier, effacer, tout ensemble de produits demandant une consultation rapide : livres, disques, relations, programmes, élèves, etc., etc.

Après le RUN, FISH vous demandera si vous voulez utiliser un fichier déjà existant sur cassette, ou si vous désirez en créer un nouveau. Dans ce cas, il vous faudra initialiser ce nouveau fichier (nombre de rubriques et titre de chaque rubrique). Le menu vous permettra, ensuite de créer de nouvelles fiches (1), de trier ces fiches dans l'ordre alphabétique des premières rubriques (2), de modifier le contenu d'une fiche (3), d'annuler une fiche (4), de lister votre répertoire à l'écran (5), ou sur une imprimante (6), d'enregistrer le nouveau fichier créé, ou modifié (7), d'effacer l'ensemble des fiches en mémoire (8) ou de sortir du programme, l'option (7), ne doit surtout pas être oubliée, en fin de traitement, il arrive, en effet, très souvent que l'on travaille plusieurs heures sur un fichier, et qu'on éteigne sa machine sans enregistrer les données ; l'option (8) est à utiliser avec prudence car elle efface toutes les fiches en mémoire.

Ce programme peut tourner sur MZ-700 mais peut être entré sur tout MZ, il est standard à tous les BASIC une exception près.

POKE $55, XX deviendra POKE 4466,XX en BASIC 5025 (MZ-80K) ou CURSOR 0,XX sur les autres BASIC.

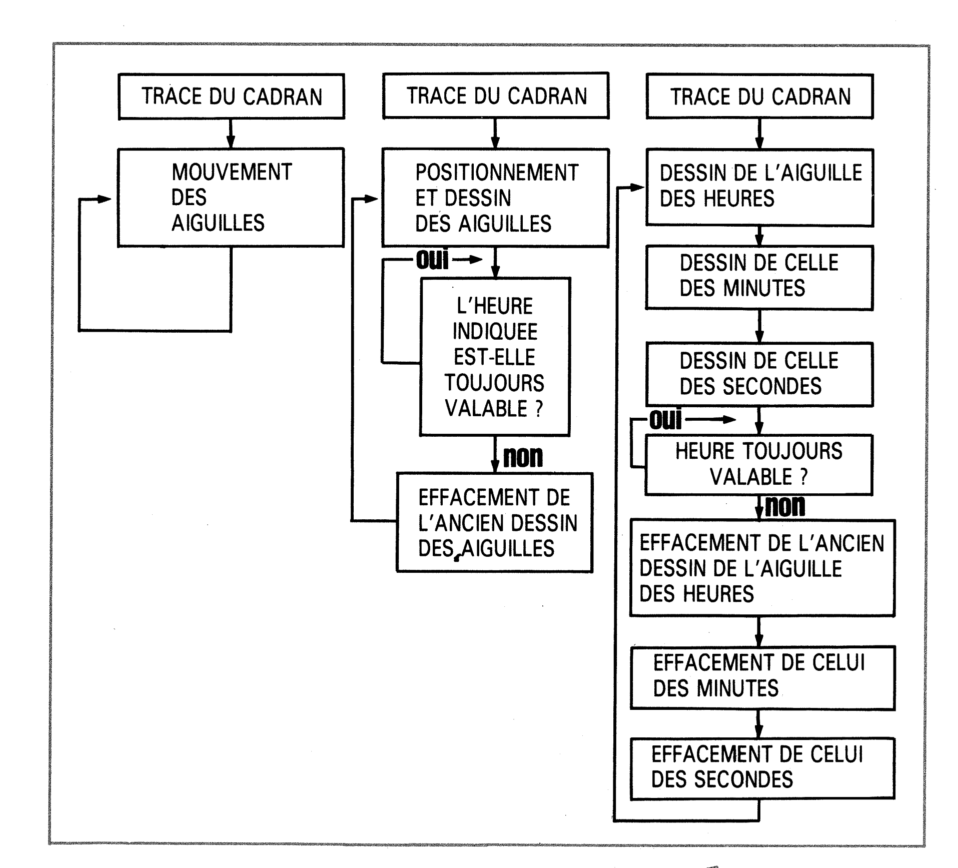
```
10 GOTO2860
20 REM ♦♦♦♦♦ TRI ♦♦♦♦♦
30 PRINT" Je trie ..."
40 S=1:SL(1)=1:SR(1)=EN
50 Q=SL(S):R=SR(S):S=S-1
60 X=Q:J=R:X$=N$(1,INT((Q+R)/2))
70 IFN$(1,X)<X$THENX=X+1:GOTO70
80 IFX$<N$(1,J)THENJ=J-1:GOTO80
90 IFX>JTHEN130
100 FORI=1TON
110 O$(I)=N$(I,X):N$(I,X)=N$(I,J):N$(I,J)=O$(I):NEXTI
120 X=X+1:J=J-1
130 IFX<=JTHEN70
140 IFX>=RTHEN160
150 S=S+1:SL(S)=X:SR(S)=R
160 R=J
170 IFQ<RTHEN60
180 IFS<>0THEN50
190 GOTO360
200 REM ♦♦♦♦♦ LECTURE DONNEES K7 ♦♦♦♦♦
210 INPUT" TITRE DU FICHIER : ";F$
220 PRINT"Placez votre cassette fichier; appuyez  sur PLAY puis tapez [ESPACE]
":REM UNIQUEMENT POUR MZ 700
230 GETA$:IFA$=" "THEN250
240 GOTO230
250 PRINT" Je lis ";F$
260 PRINT:ROPENF$
270 INPUT/TN,L:DIMC$(N):FORI=1TON:INPUT/TC$(I):NEXTI
280 DIMN$(N,255),O$(N),D$(255),W$(255)
290 INPUT/TEN:FORX=1TOEN
300 FORI=1TON:INPUT/TN$(I,X):NEXTI
310 NEXTX:CLOSE:GOTO330
320 DIMN$(N,255),O$(N),D$(255),W$(255)
330 P$=LEFT$(SL$,L)
340 L$="└"+LEFT$(SL$,L)+"┘"
350 REM ♦♦♦♦♦ MENU ♦♦♦♦♦
360 D1=0:D2=0:D3=0
370 PRINT"────────────────────────────────";
380 PRINT" ";F$;TAB(27);EN;" ENTREES"
390 PRINT"────────────────────────────────"
400 PRINT" Tapez [1] pour ENTRER DES DONNEES"
410 PRINT"        [2] pour TRIER DES DONNEES"
420 PRINT"        [3] pour MODIFIER DES DONNEES"
430 PRINT"        [4] pour EFFACER DES DONNEES"
440 PRINT"        [5] pour UNE LISTE A L'ECRAN"
450 PRINT"        [6] pour EDITER SUR IMPRIMANTE"
460 PRINT"        [7] pour ENREGISTRER DES DONNEES"
470 PRINT"        [8] pour EFFACER LES DONNEES"
480 PRINT"        [9] pour SORTIR DU PROGRAMME":USR(62)
490 PRINT" Memoire disponible :";SIZE;" octets."
500 GETA:IFA=0THEN500
510 ONAGOTO530,30,1180,1110,1340,1610,2690,2780,2840
520 REM ♦♦♦♦♦ ENTREE DES DONNEES ♦♦♦♦♦
530 IFEN>254THENPRINT" Fichier plein":USR(62):FORDL=1TO2000:NEXTDL:GOTO360
540 X=EN
550 X=X+1
560 PRINT" Tapez [-] si vous passez cette entree."
570 PRINT" Entree No.";X
580 FORI=1TON
590 POKE$55,2+2*I:PRINT"*** ";C$(I);" :":PRINT"";L$
600 POKE$55,3+2*I:INPUT" ";N$(I,X)
610 IFN$(I,X)="-"THENN$(I,X)=""
620 IFN$(I,X)="0"THENEN=X-1:GOTO360
630 IFLEN(N$(I,X))>LTHENPOKE$55,3+2*I:PRINTSPC(39):GOTO590
640 POKE$55,4+2*I:PRINTSPC(39)
650 NEXTI
660 PRINT"Toutes les entrees sont correctes ?(O/N)":USR(62)
670 GETA$:IFA$="O"THEN700
680 IFA$="N"THENGOSUB1200:PRINT"":GOTO700
690 GOTO670
700 PRINT"   Y a t-il d'autres entrees ? (O/N)     ":USR(62)
710 GETA$:IFA$=""THEN710
720 IFA$="O"THEN550
730 IFA$="N"THENEN=X:GOTO360
740 GOTO710
750 REM ♦♦♦♦♦ RECHERCHE ♦♦♦♦♦
760 PRINT" Tapez le No. d'entree ou ";C$(1)
770 PRINT" [0] Pour retourner au menu."
780 INPUT" ";DT$:PRINT:IFDT$="0"THEND1=0:D2=0:GOTO360
790 AD=ASC(LEFT$(DT$,1)):IF(AD<49)+(AD>57)THENGOSUB880:RETURN
800 X=VAL(DT$):IFX>ENTHEN760
810 IF(D1=0)*(D3=0)THENRETURN
820 PRINT"Entree No.";X:PRINT:FORI=1TON:PRINTN$(I,X):NEXTI:D1=0
830 IFD2=0THENRETURN
840 PRINT" Je l'efface vraiment ? (O/N)"
850 GETA$:IFA$="O"THENRETURN
860 IFA$="N"THEN760
870 GOTO850
880 V=0:X=1
890 IFN$(1,X)=DT$THENV=V+1:H(V)=X
900 IFX>ENTHENX=EN:GOTO920
910 X=X+1:GOTO890
920 IFV=0THEN990
930 IFV=1THENX=H(V):D1=0:GOTO810
940 PRINT" No.":FORX=1TOV:P=H(X)
950 PRINTP;" ";N$(1,P);", ";N$(2,P):NEXTX:PRINTSL$
960 INPUT"Donnez le No. (0 si pas de No.) : ";X:IFX=0THEN760
970 IFX>ENTHENPRINT"";SPC(40);"":GOTO960
980 GOTO810
990 PRINT" Je n'ai pas trouve cette entree."
1000 FORDD=1TO1000:NEXT
1010 X=1
1020 IFLEFT$(N$(1,X),3)=LEFT$(DT$,3)THENV=V+1:H(V)=X
1030 IFX>ENTHENX=EN:GOTO1050
1040 X=X+1:GOTO1020
1050 IFV=0THENUSR(62):GOTO760
1060 PRINT" Etes-vous sur de votre demande ?"
1070 PRINT" Les entrees similaires sont :":IFV>1THEND1=0:GOTO940
1080 FORDD=1TO1000:NEXT:D1=0
1090 X=H(V):GOTO810
1100 REM ♦♦♦♦♦ EFFACEMENTS ♦♦♦♦♦
1110 D1=1:D2=1:D3=1:GOSUB760
1120 PRINT" Patience....."
1130 G=X:FORI=1TON:N$(I,G)="":NEXTI
1140 FORX=GTOEN:FORI=1TON
1150 N$(I,X)=N$(I,X+1):NEXTI:NEXTX
1160 EN=EN-1:GOTO1110
1170 REM ♦♦♦♦♦ MODIFICATIONS ♦♦♦♦♦
1180 GOSUB760
1190 GOSUB1200:GOTO1180
1200 PRINT"  ┌";P$;"┐"
1210 FORI=1TON:POKE$55,I:PRINTI;" ";N$(I,X):NEXTI
1220 PRINTSL$
1230 PRINT"No. de ligne a changer (0=aucune): "
1240 GETI$:IFI$=""THEN1240
1250 IFI$="0"THENRETURN
1260 I=VAL(I$):IF(I=0)+(I>N)THEN1240
1270 PRINT" Tapez [-] pour effacer la ligne"
1280 PRINT" sinon [CR] apres correction."
1290 POKE$55,I:INPUT">";N$(I,X)
1300 IFLEFT$(N$(I,X),1)="-"THENN$(I,X)=""
1310 N$(I,X)=LEFT$(N$(I,X),L)
1320 GOTO1200
1330 REM ♦♦♦♦♦ LISTE ECRAN ♦♦♦♦♦
1340 PRINT" Tapez [1] pour ";C$(1)
1350 PRINT"        [2] pour la liste complete"
1360 GETA:IF(A=0)+(A>2)THEN1360
1370 PRINT"Tapez [1] pour une liste CONTINUE.
1380 PRINT"       [2] pour une liste PAS A PAS."
1390 GETB:IF(B=0)+(B>2)THEN1390
1400 INPUT" No. de DEPART : ";C
1410 IF(C<1)+(C>N-1)THEN1400
1420 IFB=2THEN1450
1430 PRINT"Tapez [ESPACE] pour interrompre la liste"
1440 PRINT"       [0]       pour retourner au menu.":FORDD=1TO2000:NEXT
1450 PRINT"";:FORX=CTOEN:PRINT"Entree No.";X:PRINT"──────────────"
1460 PRINTN$(1,X):IFA=1THEN1480
1470 FORI=2TON:PRINTN$(I,X):FORJ=1TO150:NEXTJ,I
1480 IFB=2THEN1520
1490 GETB$:IFB$="0"THEN360
1500 IFB$=" "THENFORDD=1TO500:NEXT:GOTO1520
1510 GOTO1570
1520 IFB=2THENPRINT" Tapez [ESPACE] pour l'entree suivante"
1530 GETA$:IFA$=" "THEN1560
1540 IFA$="0"THEN360
1550 GOTO1530
1560 IFB=2THENPRINT"";
1570 PRINT:NEXTX
1580 PRINT" FIN. Tapez "0" pour le menu.":GOTO1530
1590 REM ♦♦♦♦♦ IMPRIMANTE ♦♦♦♦♦
1600 REM Pour imprimante 1P01 (MZ 730)
1610 PRINT"";:A1=0:A2=0:A3=0:A4=0:A5=0:A6=0
1620 FORI=1TOEN:W$(I)="":D$(I)="":NEXTI
1630 DIMB$(8),B(8):FORI=1TO8:B$(I)="":B(I)=0:NEXTI
1640 PRINT"Tapez [1] pour imprimer toutes les":PRINTTAB(11);"entrees"
1650 PRINT"      [2] pour imprimer les entrees de":PRINTTAB(11);"votre choix."
1660 GETA1:IF(A1=0)+(A1>2)THEN1660
1670 PRINT" Tapez [1] pour imprimer par LABELS"
1680 PRINT"       [2] pour imprimer en CONTINU"
1690 GETA2:IF(A2=0)+(A2>2)THEN1690
1700 IFA2=1THEN1850
1710 PRINT" Tapez [1] pour UNE  rubrique par ligne"
1720 PRINT"       [2] pour DEUX rubriques par ligne"
1730 GETA3:IF(A3=0)+(A3>2)THEN1730
1740 IFA3=1THEN2210
1750 PRINT" Les rubriques pour ";F$;" sont :"
1760 FORI=1TON:PRINTI;" - ";C$(I):NEXTI
1770 PRINT:FORI=1TONSTEP2
1780 PRINT"Ligne";I;": RUBRIQUE No.: ";
1790 GETB(I):IF(B(I)<1)+(B(I)>N)THEN1790
1800 PRINTC$(B(I)):IFI+1>NTHENI=N:NEXTI:GOTO2210
1810 PRINT"Ligne";I;": RUBRIQUE No.: ";
1820 GETB(I+1):IF(B(I+1)=0)+(B(I+1)>2)THEN1820
1830 PRINTC$(B(I+1)):NEXTI
1840 GOTO2210
1850 PRINT" 8 lignes MAXIMUM peuvent etre imprimees"
1860 PRINT"pour chaque LABEL."
1870 PRINT" Voulez-vous imprimer ";C$(1);" et"
1880 PRINTC$(2);" sur la meme ligne ? (O/N)"
1890 GETA4$:IFA4$="N"THENA4=1:GOTO1950
1900 IFA4$="O"THENA4=2:GOTO1920
1910 GOTO1890
1920 PRINT" Tapez [1] pour ";C$(1);" en premier."
1930 PRINT"       [2] pour ";C$(2);" en premier."
1940 GETA5:IF(A5=0)+(A5>2)THEN1940
1950 A7=0:PRINT" Les rubriques de ";F$;" sont : "
1960 PRINT:FORI=1TON:PRINT" Rubrique ";I;": ";C$(I):NEXTI
1970 PRINT" Entrez le No. de rubrique pour chaque"
1980 PRINT"ligne du label:"
1990 PRINT" Ligne 1: ";:IF(A4=1)+((A4=2)*(A5=1))THENPRINTC$(1);:D$=C$(1)
2000 IF(A4=2)*(A5=2)THENPRINTC$(2);" ";C$(1):D$=C$(2)+" "+C$(1)
2010 IF(A4=2)*(A5=1)THENPRINT", ";C$(2):D$=C$(1)+", "+C$(2)
2020 PRINT
2030 PRINT" Tapez [-] pour entrer une ligne vide.":PRINT
2040 FORI=2TO8:PRINT"Ligne";I;": ";
2050 GETB$(I):IFB$(I)=""THEN2050
2060 IFB$(I)="-"THENB(I)=0:GOTO2080
2070 B(I)=VAL(B$(I)):IF(B(I)=0)+(B(I)>N)THEN2050
2080 PRINTC$(B(I)):A7=A7+1:NEXTI
2090 IFA7>8THEN1950
2100 PRINT" Ligne 1: ";D$
2110 FORI=2TO8:PRINT" Ligne";I;": ";C$(B(I))
2120 NEXTI
```

```
2130 PRINT"■ D'accord ? (O/N)"
2140 GETA$:IFA$="N"THEN1950
2150 IFA$="O"THEN2170
2160 GOTO2140
2170 PRINT"■ Tapez [1] pour un format normal."
2180 PRINT"■          [2] si vous voulez la derniere"
2190 PRINT"               ligne margee a droite."
2200 GETA6:IF(A6=0)+(A6>2)THEN2200
2210 IFA1=1THENGOSUB2620
2220 IFA1=2THENGOSUB2480
2230 PRINT"■ Preparez votre imprimante et"
2240 GOSUB2650
2250 IFA2=1THEN2340
2260 MODETL:PRINT/PF$;:MODETN
2270 FORX=CTOEN
2280 IF(A1=2)*(W$(X)="")THEN2330
2290 PRINT/P:FORI=1TON
2300 IFA3=1THENPRINT/PN$(I,X)
2310 IFA3=2THENPRINT/PN$(B(I),X);" ";:IFI/2=INT(I/2)THENPRINT/P
2320 NEXTI
2330 NEXTX:PRINT/P:GOTO360
2340 PRINT/P"";
2350 PRINT/P"";
2360 FORX=CTOEN
2370 IF(A1=2)*(W$(X)="")THEN2470
2380 IFA4=1THEND$(X)=N$(1,X)
2390 IFA5=1THEND$(X)=N$(1,X)+", "+N$(2,X)
2400 IFA5=2THEND$(X)=N$(2,X)+" "+N$(1,X)
2410 IFLEN(D$(X))>32THENPRINT/P"■";
2420 PRINT/PD$(X):IFLEN(D$(X))>32THENPRINT/P"■";
2430 FORH=2TO7:PRINT/PN$(B(H),X):NEXTH
2440 IFA6=2THENPRINT/PSPC(39-LEN(N$(B(8),X)));" ";
2450 PRINT/PN$(B(8),X)
2460 PRINT/P
2470 NEXTX:GOTO360
2480 PRINT"■ IMPRESSIONS SELECTIONNEES
2490 PRINT"──────────────────────────────"
2500 PRINT"■ Quand chaque entree est affichee,"
2510 PRINT"■ Tapez [1] pour l'imprimer."
2520 PRINT"■        [ESPACE] pour passer."
2530 GOSUB2620
2540 FORX=CTOEN
2550 PRINT"■ Fiche No.";X:PRINT:PRINT:FORI=1TON:PRINTN$(I,X):NEXTI
2560 GETW$(X):IFW$(X)=""THEN2560
2570 IFW$(X)="0"THEN360
2580 IFW$(X)=" "THENW$(X)="":GOTO2610
2590 IFW$(X)="1"THEN2610
2600 GOTO2560
2610 NEXTX:RETURN
2620 INPUT"■ No. de FICHE de DEPART : ";C
2630 IF(C=0)+(C>EN-1)THEN2620
2640 RETURN
2650 PRINT"■ Tapez [S] pour l'impression."
```

```
2660 GETA$:IFA$="S"THENRETURN
2670 GOTO2660
2680 REM ♦♦♦♦♦ SAUVEGARDE DES DONNEES
2690 PRINT"■ Placez votre cassette de donnees."
2700 PRINTTAB(18);"■■└─────────────┘■■"
2710 INPUT"Titre dU fichier : ";F$:F$=LEFT$(F$,16):PRINT
2720 PRINT"■ Appuyez sur RECORD-PLAY":PRINT"■ J'enregistre ";F$:REM UNIQUEMENT P
OUR MZ 700
2730 WOPENF$
2740 PRINT/TN,L:FORI=1TON:PRINT/TC$(I):NEXTI
2750 PRINT/TEN:FORX=1TOEN
2760 FORI=1TON:PRINT/TN$(I,X):NEXTI
2770 NEXTX:CLOSE:GOTO360
2780 REM ♦♦♦♦♦ EFFACE LE FICHIER ♦♦♦♦♦
2790 PRINT"■■ J'efface tout le fichier O.K ? (O/N)"
2800 GETRE$:IFRE$="O"THENRUN
2810 IFRE$="N"THEN360
2820 GOTO2800
2830 REM ♦♦♦♦♦ EFFACE LE PROGRAMME ♦♦♦♦
2840 PRINT"■■ FISH est termine.■":END
2850 REM ♦♦♦♦♦ TITRE ET INITIALISISATION
2860 PRINT"■■■■"
2870 PRINT"                  ▒▒▒ ▒ ▒▒▒ ▒ ▒"
2880 PRINT"                  ▒   ▒ ▒   ▒ ▒"
2890 PRINT"                  ▒▒  ▒ ▒▒▒ ▒▒▒"
2900 PRINT"                  ▒   ▒   ▒ ▒ ▒"
2910 PRINT"                  ▒   ▒ ▒▒▒ ▒ ▒"
2920 PRINT"■■";TAB(12)"GESTION DE FICHIER"
2930 M=12:DIMSL(M),SR(M),H(100):X=0:EN=0
2940 SL$="────────────────────────────────────────"
2950 SP$="                                        "
2960 PRINT"■■■■■Tapez [1] pour creer un nouveau fichier."
2970 PRINT"■        [2] pour lire un fichier sur K7.":USR(62)
2980 GETA:IF(A=0)+(A>2)THEN2980
2990 IFA=2THEN210
3000 REM ♦♦♦♦♦ INITIALISATION FICHIER
3010 PRINT"■";TAB(19);"■└─────────────┘":USR(62)
3020 INPUT"■ Titre du fichier : ";F$:F$=LEFT$(F$,15)
3030 PRINT"■■ Nombre de RUBRIQUES (2-9): ";
3040 GETN:IFN<2THEN3040
3050 PRINTN:DIMC$(N)
3060 INPUT"■Longueur MAXI d'une RUBRIQUE (4-37): ";L
3070 IF(L<4)+(L>37)THENPRINT"■";SPC(39);"■■":GOTO3060
3080 FORI=1TON
3090 POKE$55,5+2*I:PRINT"■";TAB(22);"└─────────────┘■■"
3100 PRINT"NOM de la RUBRIQUE";I;" :";:INPUT" ";C$(I)
3110 NEXTI
3120 PRINT"■■ Titre : ";F$:FORI=1TON
3130 PRINT"■  Rubrique";I;": ";C$(I):NEXTI
3140 POKE$55,23:PRINT" Tout est correct ? (O/N)■":USR(62)
3150 GETA$:IFA$="O"THEN320
3160 IFA$="N"THEN3010
3170 GOTO3150
```

PROGRAMMATION STRUCTUREE

Que tous les fanatiques de la programmation structurée se rassurent, si cette rubrique était absente du dernier numéro c'était à cause de la conjonction maléfique Vacances-Sicob qui perturbe toujours un peu la rédaction du bulletin de la rentrée.
Nous reprenons donc cette série, d'autant plus que M. KOKANOWSKY a récidivé en nous écrivant un SUPER BASIC pour le MZ-700 encore plus performant que le SUPER BASIC 2 !!!



Les deux premiers chapitres avaient mis l'accent sur le gain en lisibilité et en facilité de programmation que provoquait l'abandon des GOTOS au profit de boucles structurées. Nous allons voir aujourd'hui un autre avantage de ce « style » de programmation : la procédure. Prenons un exemple : vous avez envie d'utiliser votre MZ comme une horloge à aiguilles. Comme ce merveilleux appareil comporte une horloge intégrée (TI$) le problème se résume au dessin du cadran et au mouvement des aiguilles et l'algorythme de programmation peut-être représenté par n'importe lequel de ces schémas dont seul change le degré de l'analyse.

On s'aperçoit en regardant le schéma le plus détaillé que l'on effectue trois traitements très proches :

- dessin des aiguilles des heures
- dessin des aiguilles des secondes
- dessin des aiguilles des minutes

Or les seules différences entre ces trois traitements sont ;

- la position de l'aiguille
- sa longueur (en général l'aiguille des heures d'une pendule est plus petite).

De plus effacer un objet sur l'écran d'un ordinateur consiste simplement à le dessiner de la même couleur que le fond de l'écran. Il serait donc bien agréable, plutôt que d'écrire six morceaux de programme très voisins, de disposer d'un sous-programme, (on parle de procédure de subroutine, ou de routine, quand on jargonne dans le patois des informaticiens) qu'on appellerait quand on voudrait et qui dessinerait une aiguille, soit de la couleur du fond, soit normalement à une position donnée, d'une longueur choisie. (Ces trois indications à fournir au sous-programme s'appellent des paramètres).

Et avec le SUPER BASIC ou K BASIC ce rêve devient réalité !

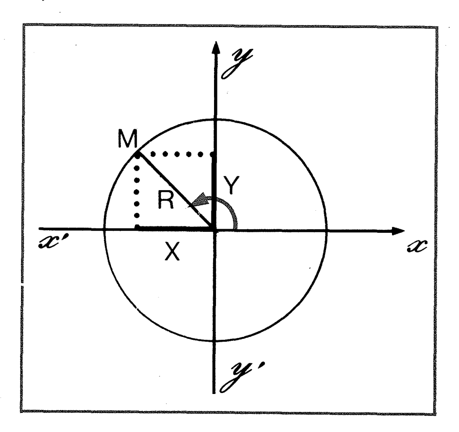
Il suffit de définir le sous-programme par DEF PROC, et de le terminer par END PROC, et on peut l'appeler six fois par PROC « nom » (paramètre 1, paramètre 2, paramètre 3, etc.). Dans notre exemple, le nom de la procédure est « aiguille » et elle comporte 3 paramètres, position, longueur et existence).

EXPLICATION DU PROGRAMME

Rappel pour les non matheux : un point M situé à la circonférence d'un cercle de rayon R et de centre O confondu avec celui du repère a comme coordonnées :

Xx = R(COS(O))

Yy = R(SIN(O))



Ligne 60-80 :
Cette boucle parcourt le cercle trigonométrique en 12 étapes pour afficher les chiffres du cadran.

Ligne 120-140 :
On découpe la variable système TI$ qui contient l'heure en trois variables numériques une pour les heures, une pour les minutes, une pour les secondes.

Ligne 150-170 :
On appelle trois fois la procédure pour dessiner les aiguilles.

Note : On multiplie les heures par 5 (entre 0 et 12) car la procédure attend une position comprise entre 0 et 60.

Ligne 180-210 :
On stocke l'absence actuelle (c'est-à-dire celle affichée) et on boucle tant qu'elle correspond à l'ancienne boucle (TI$).

Ligne 220-240 :
On efface les aiguilles (3e paramètre, celui d'« existence » à zéro).

Ligne 280 :
Les variables TEMPS, LONGUEUR, EXISTENCE prendront la valeur des paramètres qui sont envoyés à la procédure.

Ligne 300 :
Pour convertir la position de l'aiguille de 0 à 60 en une mesure, en radian ce fait :

- une règle de 3

$$\frac{\text{Temps} \times 2\pi}{60}$$

- une complémentaire (le SENS trigo est l'inverse de celui des aiguilles d'une montre)
- une rotation d'1/4 de tour ($\pi/2$).

PROGRAMME HORLOGE

```
20 PRINT"■      ENTREZ L'HEURE SOUS LA FORME HHMMSS":PRINT:PRINT
30 INPUT"HEURE:";A$:TI$=A$
40 PRINT"■      TAPEZ UN ESPACE POUR ARRETER"
50 REM ******* CADRAN *********
60 FOR THETA=π/2 TO 5*π/2 STEP 2*π/12
70 CURSOR20-COS(THETA)*10,12.5-SIN(THETA)*10
80 PRINTHEURE:HEURE=HEURE+1
90 NEXT THETA
100 REM ******* AIGUILLES ********
110 REPEAT
120 :       H=VAL(MID$(TI$,1,2))
130 :       M=VAL(MID$(TI$,3,2))
140 :       S=VAL(MID$(TI$,5,2))
150 :       PROC"aiguille"(H*5,10,1)
160 :       PROC"aiguille"(M,15,1)
170 :       PROC"aiguille"(S,15,1)
180 :       HEURACT$=TI$
190 :               WHILE HEURACT$=TI$ DO
200 :                       KEY CARAC$
210 :               WEND
220 :       PROC"aiguille"(S,15,0)
230 :       PROC"aiguille"(M,15,0)
240 :       PROC"aiguille"(H*5,10,0)
250 UNTIL CARAC$=" "
260 END
270 REM ****** PROCEDURES *******
280 DEF PROC"aiguille"(TEMPS,LONGUEUR,EXISTENCE)
290 :  GRAPH 40,25
300 :  ANGLE=-(TEMPS*2*π/60)+π/2
310 :  DRAW EXISTENCE,40+COS(ANGLE)*LONGUEUR,25-SIN(ANGLE)*LONGUEUR
320 ENDPROC
```

L.M. SUR M.Z.

Dans la série « le L.M., ça existe aussi sur les MZ », nous allons tenter par cette série d'articles de faire connaissance avec le microprocesseur qui équipe nos MZ. Pour faire connaissance, il faut dialoguer, pour dialoguer, il faut parler le même langage.
Notre microprocesseur refusant catégoriquement de parler français, nous allons donc devoir apprendre le « langage machine », dialecte natal de notre « puce ». Le gros avantage, avec MZ c'est que, contrairement aux PC, ceux-ci sont tous équipés du même microprocesseur : le Z80 de ZILOG, le LM de ce Z80 sera donc identique pour toutes nos machines.



Système MZ 700 compatible CP.M

QU'EST-CE QU'UN MICROPROCESSEUR

C'est une minuscule boîte noire, vulgairement appelée « puce » munie de nombreuses pattes qui permettent de la connecter à l'ensemble des circuits de notre MZ. Cette puce possède un cœur appelé QUARTZ ou horloge qui fait « poum, poum » à la cadence de 1000.000 à 4000.000 de battements par seconde, à chaque battement de son cœur, le microprocesseur exécute 1 instruction d'un programme écrit en langage machine, et cela sans jamais se reposer. Cette formidable capacité de travail permet à notre microprocesseur d'avoir l'œil, en une fraction de seconde, sur les 64000 cases que composent les 64K de sa mémoire, ainsi que sur tous les périphériques qui lui sont connectés : clavier, écran, imprimante, disquettes, joysticks, etc.

COMMENT FONCTIONNE UN MICROPROCESSEUR

A l'intérieur de notre puce est logé un dictionnaire de toutes les instructions de base qu'elle peut exécuter (environ 70), chacune d'entre elles possède un code numérique de valeur 1 à 255 (en décimal). Le travail du microprocesseur va donc être de lire un code dans une case mémoire, d'exécuter l'instruction correspondante, de passer à la case mémoire suivante et de continuer ainsi.

Mais attention, car il y a un MAIS ; malgré toutes ses qualités, un microprocesseur possède un gros défaut : il est incapable d'avoir la moindre initiative personnelle. S'il rencontre un code qu'il ne connait pas, il se bloquera ou fera n'importe quoi et faire n'importe quoi 1 million de fois par seconde c'est généralement désastreux pour le programme en cours.
Rassurez-vous, toutefois, un « plantage » ne peut en aucun cas, provoquer une panne matérielle sur votre MZ.

POURQUOI LE L.M.

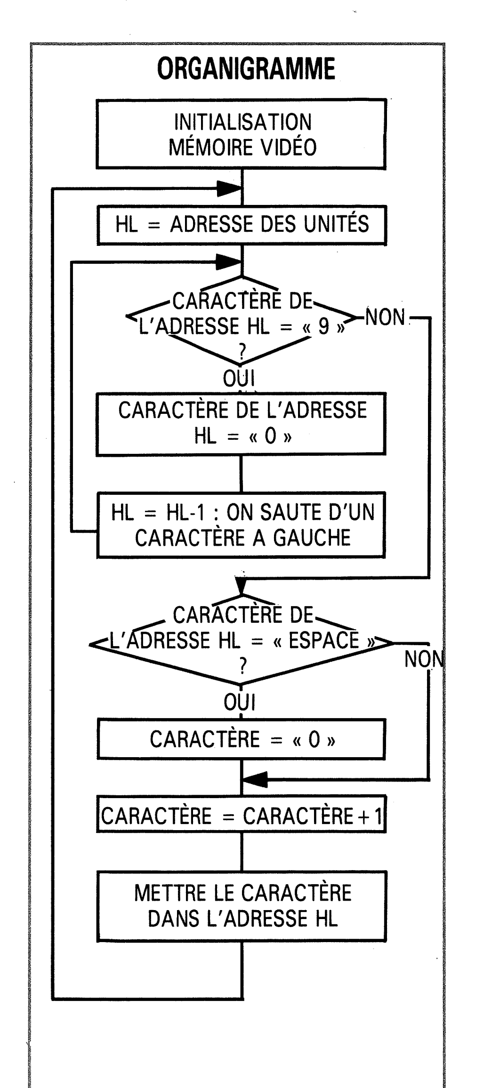
Outre sa formidable rapidité d'exécution (voir l'exemple qui va suivre), un programme écrit en L.M. est toujours beaucoup moins gourmand en place mémoire qu'un même programme en langage évolué (ex : BASIC), enfin certaines instructions du langage assembleur sont impossibles à reproduire en BASIC.

UN EXEMPLE

Sans plus entrer dans les détails, dans cette première approche, nous allons étudier, comparer et faire tourner un même programme en BASIC et en LM.

Commencez par taper le programme BASIC « COMPTE » ci-joint et ATTENDEZ la fin de ces explications avant de le faire tourner.

« COMPTE » est une programme complètement inutile mais pédagogique qui comme son nom l'indique, fait compter votre MZ de 1 à l'infini le plus rapidement possible. Pour comprendre l'organigramme qui suit, sachez que le Z80 possède plusieurs registres internes qui sont en fait des cases mémoires ou l'on peut loger des valeurs numériques. Nous utiliserons deux de ces registres appelés A et HL ; « CARACTERE » est le chiffre dont le programme calcule l'emplacement et la valeur avant de l'afficher. l'ADRESSE est la case memoire écran (voir article sur la mémoire vidéo du n° 8) ayant pour numéro : HL).



LE PROGRAMME

Etudions maintenant le programme BASIC :

COMPTE 1 est la reproduction fidèle en BASIC de la structure de notre programme LM, il vous permettra de mieux comprendre l'organigramme.

COMPTE 2, est un comptage par incrémentation d'une variable (incrément = + 1) et son affichage.

COMPTE 3, est le même comptage, mais placé dans une boucle FOR... NEXT, c'est la méthode la plus rapide en BASIC.

COMPTE 4, est notre programme L.M. qui a précédemment été implanté en mémoire par la routine « IMPLANTATION COMPTE 4 LM » les codes sont entrés par DATA'S. Sur chaque ligne de DATA'S correspond l'instruction exécutée par le Z80 (EN REM). Nous reviendrons bien sûr dans le détail sur ces instructions.

Taper RUN puis successivement 1,2,3 et 4, les différents comptages apparaîtront à l'écran, tapez [SHIFT-BREAK] pour les interrompre, ou RESET pour interrompre le programme L.M.

Comparons maintenant les temps d'exécution (MZ-700).

– En 1 minute COMPTE 1 affiche = 5100
COMPTE 2 affiche = 7100
COMPTE 3 affiche = 8500
COMPTE 4 affiche = 1.330.000 !...

COMPTE 4 est donc 260 fois plus rapide que COMPTE 1.
187 fois plus rapide que COMPTE 2.
et 156 fois plus rapide que COMPTE 3 (version la plus rapide en BASIC).

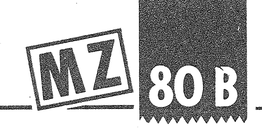
De plus COMPTE 4 occupe la place mémoire d'environ 1/2 ligne de BASIC.

Convaincus ? Alors étudiez bien ce programme et son organigramme, c'est une excellente gymnastique... La suite au prochain numéro...

S. BIZOIRRE

```
20 REM
    *********************************
30 REM*MISE EN PLACE DES INSTRUCTIONS
    *L.M. EN MEMOIRE, A PARTIR DE L'
    *ADRESSE HEXA $9000 ET MENU
    **********************************
50 GOSUB 600:GOTO1000
70 REM
    *********************************
80 REM*   COMPTE BASIC TOURNANT SELON LE
    *MEME PRINCIPE QUE LE PROGRAMME L.
    *M. (VOIR ORGANIGRAMME).
    **********************************
100 HL=$D198
110 A=PEEK(HL)
120 IF A<>$29 THEN GOTO 160
140 POKE HL,$20
150 HL=HL-1:GOTO 110
160 IF A<>$0 THEN GOTO 180
170 A=$20
180 A=A+1
190 POKE HL,A
200 GOTO 100
270 REM
    *********************************
280 REM*COMPTE AVEC INCREMENTATIONS SUC-
    *CESSIVES DE A.
    **********************************
300 CURSOR0,10:A=0
310 A=A+1:PRINT"■";A:GOTO310
370 REM
    *********************************
380 REM*COMPTE DANS UNE BOUCLE FOR.NEXT
    **********************************
400 CURSOR0,10
410 FORA=1TO99999
420 PRINT"■";A
430 NEXTA
470 REM
    *********************************
480 REM*COMPTE EN L.M.
    **********************************
500 USR($9000):REM Pour MZ 700
510 USR(36866):REM Pour MZ 80 K & A
570 REM
    *********************************
580 REM*LECTURE DES DATA'S
    **********************************
600 FORI=$9000 TO $9018
610 READ A:POKEI,A
620 NEXT I
630 RETURN
770 REM
780 REM*CODES D'INSTRUCTIONS L.M.EN DATA
    *ET INSTRUCTIONS CORRESPONDANTES
    *Z 80 EN REM.
790 REM**********************************
800 DATA $D3,$E3:     REM OUT ($E3),A
810 DATA $21,$98,$D1:REM LD  HL,$D198
820 DATA $7E:         REM LD  A,(HL)
830 DATA $FE,$29:     REM CP  $29
840 DATA $20,$05:     REM JR  NZ,$900F
850 DATA $36,$20:     REM LD  (HL),$20
860 DATA $2B:         REM DEC HL
870 DATA $18,$F6:     REM JR  $9005
880 DATA $FE,$00:     REM CP  $00

890 DATA $20,$02:     REM JR  NZ,$9015
900 DATA $3E,$20:     REM LD  A,$20
910 DATA $3C:         REM INC A
920 DATA $77:         REM LD  (HL),A
930 DATA $18,$E9:     REM JR  $9002
970 REM
    *********************************
980 REM*CHOIX DE COMPTE
    *********************************
1000 PRINT"■"
1010 PRINT" Tapez [1] Pour COMPTE 1■"
1020 PRINT"       [2] Pour COMPTE 2■"
1030 PRINT"       [3] Pour COMPTE 3■"
1040 PRINT"       [4] Pour COMPTE 4 (L.M
.)"
1050 GETRE:IFRE=0THEN1050
1060 ON RE GOTO 100,300,400,500
1070 END
```



Nous privilégierons ce mois-ci, les utilisateurs du MZ-80B, munis de disquettes et d'une imprimante, les 2 programmes proposés nécessitent, en effet, au minimum, 1 unité de disquettes.

XRAY

Rassurez-vous, XRAY n'est pas un programme de gestion de centrale nucléaire, aucune combinaison spéciale n'est exigée pour l'utiliser. XRAY est intéressant à plus d'un titre, c'est d'abord un utilitaire qui visualisera sur l'écran de votre MZ la surface physique de votre disquette et indiquera très clairement la place occupée par chaque fichier, sa longueur et son type (OBJ-TXT-BRD-BSD).

Vous vous apercevrez ainsi que certaines de vos disquettes sont organisées de manière très... désorganisée ! et pourrez ainsi en replaçant vos différents fichiers à la queue leu-leu gagner beaucoup de place sur vos disquettes. La visualisation peut également s'effectuer, de manière plus complète sur imprimante.

Instructif également, d'un point de vue didactique, l'étude de listing de XRAY, bien documenté vous permettra d'aborder l'étude du fonctionnement du système d'exploitation qui gère l'enregistrement et la lecture des fichiers sur les quelques cm² de surface magnétique qui constituent votre disquette. Bon courage...

```
 0 REM ##################################        ####   X-RAY   ### JAN. 1983 ###
 0 REM ##################################        #######  ALCO/MAST #############
       ##################################
 1 REM
 2 REM
 3 REM
 4 REM
 5 REM
 10 LIMIT $E000
 15 DEF FNH(X)=INT(X/256):DEF FNL(X)=(X-FNH(X)*256):DEF FNP(X)=INT((X/16)+.001):
DEF FNS(X)=X-16*FNP(X)
 20 D=1:TEMPO7:MUSIC"R0":DI$=STRING$(".",16)
 30 DIM PS(16,66)
 39 REM------------menu
 40 GOSUB 1000:PRINT CHR$(14);
 41 CONSOLE R,C40:PR=0:TT=0:LL=0
 42 CURSOR 10,1:PRINT"Programme  ``X-RAY''":PRINT
 44 PRINTSTRING$("■",40):GOSUB 8000
 46 CURSOR11,8:PRINT"Unite  :  FD";STR$(D)
 47 CURSOR10,6:PRINT"(Unite 0 = FIN)"
 50 CURSOR23,8:INPUT"";D$:IF D$="0" GOTO 900
 52 D=INT(VAL(D$)):IF(D>4)+(D<1)THEN MUSIC"+B":GOTO 50
 53 POKE $FFC8,D-1:USR($FFC0):USR($5D06):IF PEEK($E001)>127 THEN PRINT"DISQUETTE
 INUTILISABLE":END
 54 CURSOR 24,10:PRINT DI$:CURSOR0,10:INPUT"Titre de la disquette : ";DI$
 56 DI$=LEFT$(DI$,16)
 60 CURSOR0,12:PRINT"Impression des details (ui ou on) ?"
 62 GETZ$:IFZ$=""THEN62
 64 IFZ$="N"GOTO 100
 66 IFZ$="O"THEN PR=1:CURSOR 0,20:PRINT SPACE$(79);:GOTO 72
 68 MUSIC"+A0-A": GOTO62
 70 REM-----depart impression
 72 CURSOR1,20:PRINT"PREPARER IMPRIMANTE, PRESSER 1 TOUCHE":MUSIC"A7"
 73 GET Y$:IF Y$<>"" THEN CURSOR 1,20:PRINT SPACE$(40):GOTO 76
 74 CURSOR1,20:PRINT"PREPARER IMPRIMANTE, PRESSER 1 TOUCHE":MUSIC"R6":GOTO 72
 76 MUSIC"B0"
 80 PRINT/P STRING$("-",79): PRINT/PCHR$(18);"   `X-RAY' -  ANALYSE D'UNE DISQUE
TTE";CHR$(19)
 82 PRINT/P STRING$("-",79):PRINT/P:PRINT/PTAB(28);"Titre : ";DI$
 84 PRINT/P STRING$("-",79):PRINT/PTAB(30);"   LOCALISATION              TAILLE"
 86 PRINT/P"NOM du FICHIER       TYPE    LOCK    PISTE    SECTEUR    SECTEURS
 OCTETS"
 88 PRINT/P STRING$("-",79)
 99 REM-----------affichage masque
 100 CONSOLER,C80:GOSUB 2000
 110 SD=1:FOR II=1 TO 46:PS(0,II)=II+SD:NEXT:PS(0,II)=0:A1$="▒":AF$=A1$:CURSOR F
NP(0)+4,FNS(0)+4:PRINTA1$;:MUSIC"-B":GOSUB 500
 120 IF PEEK($E001) GOTO 200
 130 SD=48:FOR II=0 TO 66:PS(0,II)=II+SD:NEXT:FOR II=0 TO 61:PS(1,II)=66+II+SD:N
EXT: PS(1,II)=0:A1$="▒":AF$=A1$:MUSIC"+B":GOSUB 500
 199 REM-----------balayage directory
 200 ME=57376:FOR I=1 TO 63
 209 REM-----RECHERCHE CODE
 210 AF$="":A1$="":TY=PEEK(ME):IF TY=0 GOTO 290
 220 RESTORE 230:FOR II=1 TO TY:READ A1$,AF$,MU$,TY$:NEXT
 230 DATA ▒,■,+B,OBJ,▒,▒,+D,BTX,▒,●,F,BSD,▒,♦,-D,BRD,?,?,C,???,?,?,C,???,?,?,C,?
??,?,?,C,???,?,?,C,???
 250 GOSUB 400
 290 ME=ME+32:NEXT I:GOTO 700
 399 REM------------recherche de tous les secteurs concernes
 400 ON TY GOSUB 3000,3000,4000,5000
 410 PO=INT(SD/16):SO=SD-16*PO
 420 IF PR GOSUB 800
 499 REM------------affichage
 500 CURSOR FNP(SD)+4,FNS(SD)+4:PRINT A1$;:MUSIC MU$:I1=0:I2=1
 510 PS=PS(I1,I2):IF PS=0 THEN RETURN
 520 CURSOR FNP(PS)+4,FNS(PS)+4:PRINT AF$;:REM MUSIC MU$
 530 I2=I2+1:IF I2=66 THEN I2=0:I1=I1+1
 540 GOTO 510
 690 RETURN
 699 REM-----fin d'affichage
 700 CURSOR 0,0:PRINT"J'ai fini !";:MUSIC"+F-G+F-G+F-G+F-G+F-G+F-G+F-G+F-G+F-G"
 706 CURSOR 0,0:PRINT"                    ";:CURSOR64,0:PRINT TT;" secteur";
 707 IF TT>1 THEN PRINT"s"
 708 IFPR GOSUB 7000
 710 GETZ$:IFZ$=""THEN710
 720 IF Z$="I"THEN CURSOR 0,24:PRINT SPACE$(79);:PRINT/PCHR$(17):COPY/P1:PRINT/P
CHR$(16):GOSUB 2084:GOTO710
 730 IF Z$="M"GOTO 40
 735 IF Z$="F"GOTO 900
 740 MUSIC"-F+F":GOTO 710
 800 REM-----DIR/P DETAILLE
 801 REM
 810 FOR IJ=ME+1TO ME+16:IF PEEK(IJ)=13 THEN IJ=ME+16:GOTO822
 820 PRINT/PCHR$(PEEK(IJ));
 822 NEXT IJ
 830 PRINT/PTAB(20);TY$;" ";AF$;TAB(29);CHR$(42*PEEK(ME+18));TAB(38);FNP(SD);TAB
(45);FNS(SD);
 840 TA=61:X0=SO:GOSUB 6000:TA=71:X0=L:GOSUB 6000:TA=76:PRINT/P" =";:X0=INT(L/10
2.4)/10:GOSUB 6000:PRINT/P"K "
 890 RETURN
 899 REM ---- sortie du programme
 900 LIMIT MAX:CONSOLE N,C 40
 910 USR($5D06)
 998 END
 999 REM ----- ss prog langage machine
 1000 RESTORE 1010:FOR I=65472 TO 65486:READ OC:POKE I,OC:NEXT:RETURN
 1010 DATA 221,33,200,255,195,0,93,0,0,16,0,0,8,0,224
 2000 REM-----sspgm  ECRAN
 2001 REM
 2010 PRINTTAB(15);" X-RAY   Titre : ";DI$;TAB(53);"Fichiers : "
 2020 PRINT" ──────────────────────────────────────────────────────
──────────"
 2025 PRINT"Dec.▒123456789▒123456789▒123456789▒123456789▒123456789▒123456789▒123
456789"
 2030 PRINT"    ┌─────────────────────────────────────────────────
──────┐"
 2040 PRINT"  0|
      |0"
 2041 PRINT"  1|
      |1"
 2042 PRINT"  2|
      |2"
 2043 PRINT"  3|
      |3"
 2044 PRINT"  4|
      |4"
 2045 PRINT"  5|
      |5"
 2046 PRINT"  6|
      |6"
 2047 PRINT"  7|
      |7"
 2048 PRINT"  8|
      |8"
 2049 PRINT"  9|
      |9"
 2050 PRINT" 10|
      |A"
 2051 PRINT" 11|
      |B"
 2052 PRINT" 12|
      |C"
 2053 PRINT" 13|
      |D"
 2054 PRINT" 14|
      |E"
 2055 PRINT" 15|
      |F"
 2060 PRINT"    └─────────────────────────────────────────────────
──────┘"
 2070 PRINT"Hex.▒123456789ABCDEF▒123456789ABCDEF▒123456789ABCDEF▒123456789ABCDEF
▒12345"
 2080 PRINT
 2084 CURSOR 0,23:PRINT"Signes: ▒=Pistes reservees / ▒=Systeme / ▒■=OBJ / ▒▒=BTX
 / ▒●=BSD / ▒♦=BRD";
 2090 CURSOR 0,24:PRINT "COMMANDES : mpression / enu / in";
 2100 RETURN
 2999 REM----------secteurs occupes type: OBJ/BTX
 3000 L=PEEK(ME+20)+256*PEEK(ME+21):SD=PEEK(ME+30)+256*PEEK(ME+31):SO=TT
 3010 I1=0:I2=0:FOR II=0 TO L/256:PS(I1,I2)=II+SD:I2=I2+1:IF I2=66 THEN I2=0:I1=
I1+1
 3020 NEXT:IF I2=66 THEN I2=0:I1=I1+1
 3030 PS(I1,I2)=0:TT=TT+II:SO=TT-SO:LL=LL+L
 3900 RETURN
 3999 REM----------secteurs occupes type BSD
 4000 L=PEEK(ME+20)+256*PEEK(ME+21):SD=PEEK(ME+30)+256*PEEK(ME+31):SO=TT
 4010 PS(0,0)=SD:I1=0:I2=1:CURSOR 0,0:PRINT"J'etudie...";
 4020 POKE $FFC9,FNL(SD       ):POKE $FFCA,FNH(SD       ):POKE $FFCC,$01:POKE $F
FCE,$F0
 4030 USR($FFC0)
 4050 PS(I1,I2)=PEEK($F0FE)+256*PEEK($F0FF)
 4060 POKE $FFC9,PEEK($F0FE):POKE $FFCA,PEEK($F0FF)
 4070 IF PS(I1,I2)=0 THEN USR($5D06):TT=TT+1:SO=TT-SO:LL=LL+L:CURSOR 0,0:PRINT"
          ":RETURN
 4080 TT=TT+1:I2=I2+1:IF I2=66 THEN I2=0:I1=I1+1
 4090 GOTO 4030
 4999 REM----------secteurs occupes type BRD
 5000 SD=PEEK(ME+30)+256*PEEK(ME+31):L=256:SO=TT
 5010 PS(0,0)=SD:I1=0:I2=0:TT=TT+1:CURSOR 0,0:PRINT"J'etudie...";
 5020 POKE $FFC9,PEEK(ME+30):POKE $FFCA,PEEK(ME+31):POKE $FFCC,$01:POKE $FFCE,$F
0
 5030 USR($FFC0):USR($5D06):MM=61440
 5035 BL=PEEK(MM)+256*PEEK(MM+1):IF BL=0 GOTO 5080
 5040 FOR II=0 TO 15:TT=TT+1:L=L+256
 5050 I2=I2+1:IF I2=66THEN I2=0:I1=I1+1
 5060 PS(I1,I2)=BL+II
 5070 NEXT II:MM=MM+2:GOTO 5035
 5080 I2=I2+1:IF I2=66 THEN I2=0:I1=I1+1
 5090 PS(I1,I2)=0:CURSOR 0,0:PRINT"             ";:SO=TT-SO:LL=LL+L:RETURN
 5900 RETURN
 5999 REM--------cadrage a droite
 6000 PRINT/P TAB(TA-LEN(STR$(INT(X0)))-1);X0;:RETURN
 6999 REM--------impression totaux
 7000 PR=0:Z$="I":PRINT/P TAB(45);"TOTAL :";TAB(57);"-----   -----   -----"
 7010 TA=61:X0=TT:GOSUB 6000:TA=71:X0=LL:GOSUB 6000
 7020 TA=76:X0=INT(LL/102.4)/10:GOSUB 6000
 7090 PRINT/P"K":PRINT/P:PRINT/P:RETURN
 7999 REM ---- xplications
 8000 CURSOR 0,14:PRINT"Ce programme analyse l'occupation de   l'espace disquet
te fichier par fichier"
 8010 PRINT:PRINT"avec visualisation":PRINT"du `plan d'occupation' de la disquet
te."
 8020 PRINT:PRINT"Liste detaillee des fichiers":PRINT"possible sur imprimante."
 8090 RETURN
```

PC 1500

Programme « Les nombres premiers » de la page 24.

```
 1 REM ****************************************
 2 REM *                                      *
 3 REM *          NOMBRES PREMIERS            *
 4 REM *                                      *
 5 REM ****************************************
 6 REM
 7 REM
10 DATA &B5,&01,&DD,&B7,&04,&81,&01,&DD
20 DATA &0A,&A5,&78,&C0,&1A,&6A,&07,&58
30 DATA &00,&48,&00,&84,&D9,&08,&14,&D9
40 DATA &1A,&94,&DB,&18,&FB,&00,&81,&03
50 DATA &18,&FD,&40,&88,&12,&94,&8B,&04
60 DATA &04,&86,&91,&2A,&AE,&78,&C1,&9A
70 RESTORE :X=&78C0
75 FOR I=2TO 49:READ V:POKE X+I,V:NEXT I
80 INPUT "Nombre (1-255) :";Z
90 POKE X,Z:CALL X+2:Z$="Non premier"
95 IF PEEK (X+1)LET Z$="Premier"
100 PRINT Z$:GOTO 80
```

BASIC SB-6510 + +

VERSION 1.0 : F.L (1984)
Utilisation pour SHARP MZ-80B (disquette et K7).

Cette version du Basic permet l'utilisation de plusieurs fonctions importantes, fonctions absentes dans les BASIC Sharp précédents.

- touches répétitives par frappe de la touche à répéter, et appui ensuite simultanément de GRAPH et RVS.
- apparition à l'écran des touches flèches (CHR$ (1), CHR$ (2), CHR$ (3), CHR$ (4) par appui de RVS et des touches jaunes du curseur.
- apparition à l'écran du H (CHR$ (5)) par appui de RVS et de CLR HOME.
- programmation possible de 20 touches bleues par l'AUTO RUN.

PAR EXEMPLE :
RUN, LIST, CONSOLE C80, DELETE, CHRS (DIR FD1, DIR FD2, CONT, SAVE « LOAD », LOCK, UNLOCK, CONSOLE C40, GOTO, GOSUB, RETURN, NORMAL, CONSOLE S0,24,CLR C.

- curseurs jaunes motorisés par appui continu sur le curseur jaune choisi.
- accès à des fonctions spéciales : en incorporant dans AUTO RUN un c à la fin du programme.

EXEMPLE :
100 c : PRINT CHRS (6) : NEW
Ce petit c permet l'ouverture des séquences spéciales. Il est également possible de le frapper directement avant exécution du programme.
on obtient alors :

- une rémunération automatique des lignes, des GOTO et des GOSUB en frappant RENUMBER.

EXEMPLE :
RENUMBER 100,10 (renumérotation à partir du numéro 100 et avec un pas de 10). Cette renumérotation commence toujours à partir de la première ligne du programme.

- un effacement des lignes en frappant DEL

EXEMPLE :
DEL 10,20 efface les lignes 10 à 20 incluses.

- une vérification de l'exécution du programme en frappant TRON n (0²n²255) et faisant apparaître sur l'écran les numéros des lignes en cours d'exécution.

EXEMPLE :
TRON 50 permet une vérification assez rapide pour ne pas trop ralentir l'exécution du programme. TRON 255 est la vitesse la plus lente.
On peut stopper provisoirement l'exécution du programme en appuyant sur la touche curseur jaune : flèche en bas. Le programme reprend en appuyant sur une touche quelconque.
La suppression de la fonction TRON se fait en frappant OFF

- une possibilité de visualiser les variables existantes en mémoire, en frappant DUMP n (n : de 0 à 7).

EXEMPLE :
DUMP0 : toutes les variables
DUMP 1 : les fonctions
DUMP 2 : les strings
DUMP 3 : strings à 1 dimension
DUMP 4 : strings à 2 dimensions
DUMP 5 : variables numériques
DUMP 6 : variables à 1 dimension
DUMP 7 : variables à 2 dimensions.

Dans le cas de l'utilisation de programmes techniques employant des séquences importantes de POKE et PEEK, il est parfois utile de neutralier l'accès aux fonctions spéciales. Il faut alors utiliser l'ordre NORMAL, soit directement dans le programme, soit par frappe directe sur l'écran.
On retrouve les fonctions spéciales, en frappant un c.

IMPORTANT :
Ce basic SB-6510 + +, version 2.0 : F.L permet également l'exécution des programmes écrits en BASIC C.
(voir programme démonstration « MOTEUR BATEAU » joint à la disquette BASIC.

NOTA SUR L'UTILISATION DES PROGRAMMES EN BASIC C.
Avec ce Basic SB-6510 + +,
il suffit de faire LOAD « nom du programme » et RUN, ou directement RUN « nom du programme » pour pouvoir obtenir l'exécution du programme choisi.
L'arrêt par BREAK est possible, mais efface le programme en mémoire. Il faut recharger le programme pour recommencer son exécution.
Il n'y a pas de possibilité de lister le programme, ni de le sauver individuellement, ni de passer au moniteur.
Cependant, pour les programmes écrits en Basic SB-6510 + +, toutes les fonctions sont accessibles immédiatement, notamment LIST, SAVE, MON et on peut exécuter indifféremment des programmes écrits sous les verion s BASIC SB-6510 + + ou BASIC C.
Tous les programmes écrits en BASIC 5510, BASIC 5520, BASIC 6510, BASIC 6510 +, BASIC 6520, BASIC 6610 sont compatibles, sauf spécifications spéciales à ces basics : double précision, ports pour RS 232,... etc.

PAR CONVENTION, les programmes en BASIC C sont sauvés avec le titre en inversion vidéo.

ATTENTION.
Les programmes en BASIC C ne sont pas exécutables par les autres basics des séries 5000 et 6000 et la fonction LIST fait apparaître les anomalies de fonctionnement pouvant endommager la disquette.
Vous pouvez vous procurer ce BASIC, disponible grâce à l'aimable autorisation de J.P. GLUCK, utilisateur très actif du MZ80B, en nous faisant parvenir une disquette formattée (bien emballée...) et la somme de 30 F.



LA LOGITHEQUE

... ou plutôt les Logithèques, car il nous faudrait un bulletin complet pour vous présenter la liste des logiciels du Club, toutes machines confondues.

Alors nous avons dû trancher et vous proposer notre logithèque par morceaux, ou plutôt, par catégorie d'appareils.

Nous commençons par la logithèque MZ-80K/A. Dans le prochain numéro : tous les programmes pour PC-1500 (un gros morceau !).

Elle est constituée en grande partie de logiciels conçus réalisés par les SHARPENTIERS et transmis au Club. Les modalités d'accès à la logithèque répondent à deux impératifs :

• Assurer une légitime contrepartie aux concepteurs de programmes.

• En faire bénéficier tous les Sharpentiers, mêmes débutants en programmation.

Dans cet esprit, nous avons généralisé la procédure appliquée aux MZ et l'avons modifiée comme suit :

1. Vous nous faites parvenir un ou plusieurs programmes dont vous êtes l'auteur, nous vous retournons, sous 20 jours maximum un accusé de réception accompagné d'un nombre de points équivalent à la valeur de ce(s) programme(s), de (1 à 5 par programme), vous pourrez ensuite nous demander, en une seule fois, autant de programmes de la logithèque que de points acquis. Vos programmes doivent nous parvenir sur cassette, accompagnés d'une notice aussi complète que possible.

2. Vous désirez recevoir des programmes mais n'avez pas de points à votre crédit. Faites nous parvenir, dans ce cas, une cassette vierge et la somme de 15 F. par programme demandé, nous vous retournerons vos programmes sous forme de listing ou sur cassette sous 20 jours maximum. Offre limitée à 3 programmes par ENVOI.

PROGRAMMES MZ-700

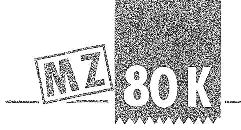
Parrallèlement à la logithèque, nous pouvons fournir aux possesseurs de MZ-700 différents programmes (jeux, utilitaires, gestion).
Sur simple demande, nous vous en adressons la liste et le tarif correspondant (la liste des logiciels MZ 700 s'est étoffée d'une dizaine de logiciels de jeux superbes).

```
LOGITHEQUE CLUB SHARP MZ 80 K/A

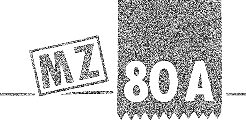
          PROGRAMMES  DE  JEUX

AH! LES FEMMES   8  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
  ALLEZ REJOINDRE L'ELUE DE VOTRE COEUR MAIS CE N'EST PAS FACILE.
AIGUILLAGE       5  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
  MANEUVRER SAVAMMENT LES AIGUILLAGES POUR EVITER TOUTE COLLISIONS.
AKRONA          12  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
  PAS TOUJOURS FACILE DE VIDER DES COLONNES (DEUX VERSIONS).
ALCOOLTEST       3  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  PAS BESOIN DE SOUFFLER DANS LE MZ  DITES SEULEMENT LA VERITE.
ALDEBARAN       22  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
  BOUCHER TOUTES LES PORTES POUR SAUVER UN MAXIMUM DE SAVANTS.
ALIEN            8  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
  EMPECHER LES EXTRA-TERRESTRES D'ENVAHIR LA PLANETE TERRE.
ALIEN FALL       5  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
  UNE AUTRE VERSION D'AKRONA (TRES BIEN EGALEMENT).
ALLIGATOR        4  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  SAUTEZ POUR ATTRAPER DES BANANES MAIS NE TOMBEZ PAS DANS LA FOSSE.
ALLUMETTES       2  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  NE PRENEZ PAS LA DERNIERE ALLUMETTE.
ALUNNISSAGE      5  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  FAITES ATTERRIR VOTRE FUSEE SANS DEGAT   (BONS GRAPH. ET SONS).
AMBULANCE        7  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
  PILOTEZ VOTRE AMBULANCE SANS AGRAVER LE SORT DE VOTRE MALADE.
ANAGRAMME        3  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
  LE MZ TROUVE ET ECRIT TOUS LES ANAGRAMMES.
ANDROMEDA GAME   8  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
  ON VOUS ATTAQUE DU CIEL IL FAUT VOUS DEFENDRE DU BAS DE L'ECRAN.
ANTARES         20  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
  A L'INTERIEUR DE L'AVION SEULE LA LIGNE DE MIRE VOUS AIDE
ARROWS           6  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
  VOUS ETES AU CIEL ON VOUS ATTAQUE DE LA TERRE
ATLANTIC PATROL  7  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
  PATROUILLE DANGEREUSE SUR LA MER.ATTENTION AUX TORPILLES.
ATTACKER        12  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
  PROMENADE HAZARDEUSE SUR UN CHAMP DE MINES !!!!.
AUTO-CROSS       4  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  TOURNEZ AUTOUR DE L'ECRAN SANS SE RETROUVER DANS LES BOTTES DE PAILLE.
BATAILLE NAVALE  9  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
  VOUS JOUEZ EN A6  MZ VOUS REPOND A L'EAU!!!.
BIORYTHME        3  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
  TOUT SUR VOTRE FORME DU MOMENT
BLOCK KUZUSHI    4  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
  JEU DE CASSE-BRIQUES TRES CONNU.
BOGGLE           5  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
  JEU DE SOCIETE DEJA CONNU.(EN TROIS DIMENTIONS).
BOMBARDEMENT     2  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  ESSAYER DE RASER LA VILLE DANS UN MINIMUM DE TEMPS.
BOULIER CHINOIS  9  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  PLUS DE SECRET SUR L'ANCETRE DES CALCULATRICE.
BOWLING          5  Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A ■ 5510 □ 5060A ■ LM □
  TRES BEAU GRAPH. POUR CE JEU CONNU.
BREAK-OUT        8  Ko. ▮K ■ 5025 □ 5060 ■ LM ■ * A □ 5510 □ 5060A □ LM □
  AUTRE CASSE-BRIQUES MAIS ON PEUT PERCER LE MUR...
BREAK-THROUGH    8  Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
  AUTRE VARIANTE DE BREAK-OUT TRES REUSSIE AUSSI.
BRIDGE           7  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  MZ JOUE CONTRE VOUS (DIFFICILE DE GAGNER).
CAMELOT          5  Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
  UNE TETE APPARAIT AUX DIFFERENTES FENETRES D'UN CHATEAU.TIREZ DEDANS!!!.
```

LISTE







CAR RACE 4 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
UNE VARIANTE DE HEAD-ON

CAR RACE SPEZIAL 6 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
EVITER LES COLLISIONS AVEC D'AUTRES VEHICULES QUI DESCENDENT ...

CAVERN DEFENDER 15 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
A L'INTERIEUR D'UN TUNNEL IL FAUT ATTAQUER MAIS AUSSI SE DEFENDRE..

CHASSE GASPIS 12 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
PARTEZ DE CHEZ VOUS SANS OUBLIER D'ETEINDRE VOS APPAREILS.

CHATEAU HANTE 20 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
JEU DE MEMOIRE.LA PARTIE PEUT DURER PLUSIEURS HEURES.

CHIFFRES ET LET. 12 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
FAUT-IL EN DONNER ?.

COLDITZ 16 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
ESSAYER DE VOUS ECHAPPER DE CETTE PRISON CELEBRE.

COSMOFIGHT 9 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
DETRUISEZ UN MAXIMUN DE PETITS METEORITES AVEC VOTRE LANCE-ROQUETTE

COUP D'ETAT 30 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
JEU DE STRATEGIE.DES BANDITS SONT DANS LA VILLE.(TRES BEAU GRAPH.)

COW BOY DUEL 13 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
COMME AU FAR-WEST SE JOUE A DEUX ON SE BLESSE ON SE TUE !!!!

CRASH 9 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
ENCORE UNE AUTRE VERSION DE HEAD-ON

DALLAS 2 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
DES COMMENTAIRES PAS TOUJOURS OBLIGEANTS POUR LA FAMILLE EWING.

DAM.SOLITAIRE 2 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
VOUS NE SEREZ JAMAIS SEUL PUISQUE VOUS ETES DEVANT VOTRE MZ.

DEFENDER 2 VOICE 16 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
SURVOL TRES DANGEREUX D'UNE MONTAGNE AUX NOMBREUX PIEGES.

DEFENDER↔B 21 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
UNE AUTRE VARIANTE DE DEFENDER TRES REUSSIE EGALEMENT.

DESAMORCAGE 3 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
LE TEMPS EST COMPTE.IL FAUT FAIRE VITE POUR TROUVER LES BOMBES.

DIGGER 84 12 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
EXPLORER TOUTE LA MINE DE CHARBON EN EVITANT LES WAGONNETS EST DUR.

DONKEY KONG 16 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
GRIMPER AUX ECHELLES.EVITER LES TONNEAUX.SAUTER DANS L'ASCENSEUR...

DOUBLE WORM 11 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
MANGER UN MAXIMUN DE CASES NUMEROTEES POUR GAGNER(SE JOUE A DEUX).

ELIMINATOR 3 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
GUERRE DE L'ESPACE.

ELISA 7 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
SI VOUS VOULEZ VOUS CONFIER A VOTRE MZ C'EST L'OCCASION OU JAMAIS.

ESCRIME 7 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
NE CRAIGNER PAS DE PERCER L'ECRAN.TOUTES (OU PRESQUE) LES FIGURES.

ESPERANCE DE VIE 5 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
LA TRISTE REALITE!!!(PUREMENT SUBJECTIF).

FLIEGE 3 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
DIRIGER VOTRE SERPENT DANS UN RECTANGLE(SANS VOUS MORDRE LA QUEUE).

FORCE 4 5 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
JEU DE STRATEGIE ET DE REFLEXION.

FROGGER TRIP 14 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
LA PAUVRE GRENOUILLE DOIT TRAVERSER LA RUE SANS DOMMAGE POUR ELLE.

GALACTICAN 14 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
SUPER SPACE INVADER.LE MEILLEUR PGM.DE JEU EN ASSEMBLEUR SUR 80 K.

GAME OF LIFE 3 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
REGARDER EVOLUER LES GENERATIONS SUCCESSIVES QUE VOUS AVEZ PROGRAMMEES

GOLF 20 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 □ 5060A ■ LM □
CHOISISSEZ VOTRE CLUB.DOSEZ VOTRE EFFORT.ET PAN! DANS LE MAGNETOPHONE.

GOMOKU MACHINE 4 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
JEU DE MORPIONS SUPER RAPIDE !!!.

GRAPHIK 5 (M) 2 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
ABSOLUMENT RIEN A FAIRE !!! TRES BELLE DEMO. DE DESSINS GEOMETRIQUES.

GRENOUILLE 7 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
ATTRAPPER UN MAXIMUM DE MOUCHERONS AVEC LA LANGUE.(AMUSE LES ENFANTS).

GUILLAUME TELL 4 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
DECOCHER UNE PLUIE DE FLECHE POUR TRANSPERCER POMMES.SERPENT ETC...

GUN GAME 7 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
UN AVION TRAVERSE L'ECRAN . A VOUS DE TIRER DESSUS...

HANOI 3 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
JEU DE REFLEXION.DISPOSEZ D'UNE MANIERE DECROISSANTE DES ANNEAUX .

HAWAII 7 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
ENFIN RIEN A FAIRE!JUSTE A PLEURER SUR LE SORT DE CES DEUX VACANCIERS.

HEAD-ON 3 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
POURSUITE INFERNALE A L'INTERIEUR D'UN LABYRINTHE.

HELL DIVER 12 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
EMPECHEZ UN CURIEUX ENTONNOIR DE SE REMPLIR.

ICC-SUPERWURM 7 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
TOTALISER UN MAXIMUM DE POINTS EN PERCUTANT DES CASES NUMEROTEES.

JUMPING BALL 3 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
IL N'Y A QU'UNE PLACE DE LIBRE ET IL FAUT REMETTRE LES PIONS DANS L'ORDRE

KINGDOM 6 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
MENER A BIEN VOTRE SEPTENAT SUR UNE ILE SINON PAS DE CADEAUX POUR VOUS!.

KRIES DER STERNE 6 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
UNE LIGNE DE MIRE AU MILIEU DE L'ECRAN.TIREZ SUR L'ENNEMI...

KRYPTON 10 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
DES METEORITES S'ABATTENT SUR VOUS.IL FAUT SE DEFENDRE.

LABYRINTHE 5 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
ESSAYER DE TROUVER LA SORTIE DANS CE LABYRINTHE A TROIS DIMENSIONS.

LUNAR TRANSPORT 3 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
TRANSPORTER DES HOMMES SUR LA LUNE N'EST PAS UNE MINCE AFFAIRE..

M-CHECKER 30 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
JEU DE DAMES MAIS REGLE DU JEU A L'ANGLAISE.

MANEGE INFERNAL 7 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
VOUS ETES AU MILIEU D'UN MANEGE ET IL FAUT TIRER SUR LES PETITS CHEVAUX.

MASTERMIND 3 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
CE N'EST PAS EN COULEURS MAIS LE PRINCIPE RESTE LE MEME.

MISSILE COMMAND 13 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
UNE D.C.A. EN MINIATURE (HEUREUSEMENT !!!!).

MONOPOLY 14. Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
EXACTEMENT COMME LE VRAI MAIS SANS GRAPH.

MORSE 2 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
POUR APPRENDRE CE CODE;PAS DE PROBLEME.

MOSAIQUE 4 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
RECONSTITUEZ UN DESSIN VU AU DEPART.TRES BIEN POUR LES ENFANTS.

MUNCHIES 7 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
UN TRES BEAU PACMAN.

OIL FIELD 16 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
ATTENTION AUX DIFFERENTS PIEGES DANS CE CHAMPS BIZARRE....

PAC-MAN IV 7 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
PLUSIEURS VITESSES POUR CE PACMAN.

PATROL ALPHA 12 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
LA CONDUITE DE CE TANK EST DELICATE SUR UN TERRAIN TRES ACCIDENTE.

PCG WARP I 14 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
ABATTEZ L'ENNEMI AVEC DEUX TIRS SIMULTANES.

PENDU 3 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
SANS COMMENTAIRE POUR CE JEU CONNU;SI CE N'EST QU'IL PLAIT AUX ENFANTS.

PHENIX 16 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
L'ENNEMI VIENT DU CIEL .A VOUS DE VOUS DEFENDRE.

PING-PONG 8 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
L'ECRAN DU MZ DEVIENT UNE TABLE IL N'Y A PAS DE FILET.(SE JOUE A DEUX).

POINTGET 6 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
POURSUITE DANS UNE GRILLE (GENRE DE PACMAN).

POKER STRIP 4 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
VOUS VERREZ DES CARTES ET AUSSI ....(CLASSE X).

POULE & RENARD 5 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
SUR UN DAMIER NE VOUS FAITES PAS CROQUER PAR CE RENARD.

PUCKMAN 16 18 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
SUPER PACMAN.PLUSIEURS JOUEURS PEUVENT S'INSCRIRE.

PUCKMEN VOICE 13 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
UN AUTRE PACMAN ET EN PLUS ON VOUS PARLE ...EN ALLEMAND.

PUISSANCE 4 6 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
JEU DE SOCIETE CONNU.SE JOUE A DEUX.

QUADRILETTRE 2 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
REMETTEZ UN ENSEMBLE DE QUATRE LETTRES DANS L'ORDRE.(PAS EVIDENT!!)

RALLEY FOR 2 8 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
CAR RACE A DEUX JOUEURS SUR LE MEME TERRAIN.

RALLEY X 11 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
CIRCULATION EMBOUTEILLEE DANS UN LABYRINTHE VISUALISE A L'ECRAN.

RAUM-SLALOM 9 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
TRAVERSEZ DES CHICANES TOUT EN VOUS PRESERVANT DE L'ENNEMI.

RED ALERT VOL II 8 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
DES GROS ENGINS FONDENT SUR VOUS.IL FAUT LES DETRUIRE A TOUT PRIX.

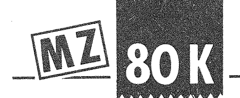
RETOURNE 6 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
EXCELLENT POUR TESTER VOTRE MEMOIRE VISUELLE.(CARTES RETOURNEES).

REVERSI 4 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
ENCORE UN OTHELLO.

ROBOFIGHT 15 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
VOUS ETES SANS CESSE ATTAQUER DANS UN DEDALE.

ROULETTE 4 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
LES ROUGES!!LES NOIRS!! IL NE MANQUE PLUS QUE L'ODEUR DES CIGARES.

RUBICUBE 6 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
TOURNEZ RETOURNEZ LE CUBE ET PEUT-ETRE VOUS GAGNEREZ.





RUN OUT 6 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
ALLEZ PERCUTER LES TETES DE MONSTRES EN HAUT DE L'ECRAN.

SARGON 2.5 +BELL 19 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
TRES BON JEU D'ECHEC.

SCRAMBLE 3 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
NAVIGUER EN TOUS SENS VOTRE VAISSEAU POUR REUSSIR LA MISSION.

SCRAMBLER 6 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
IDEM A SCRAMBLE MAIS L'ENSEMBLE EST BEAUCOUP PLUS GROS.

SKY CHIP 8 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
L'ENNEMI EST EN BAS DE L'ECRAN.VOUS ATTAQUER D'EN HAUT.(CA CHANGE...).

SOS ISLAND 14 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
VOUS ETES SUR UNE PLATEFORME PETROLIFERE ET L'AVENTURE COMMENCE.

SPACE GUERILLA 16 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
DES EXTRA-TERRESTRES ESSAIENT DE VOLER VOTRE BIEN.IL FAUT INTERVENIR.

SPACE INVADER 10 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
L'UN DES PLUS VIEUX JEU D'ARCADES SUR LES MZ.

SPACE INVADER II 3 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
VARIANTE AMUSANTE DE L'ANCETRE!!!.

SPACE PANIC 21 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
FAITES TOMBER DES VILAINS MONSTRES DANS DES TROUS CREUSES PAR VOUS..

SPEICHER-ORGEL 2 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
TOUT SUR LA MUSIQUE.JOUER ENREGISTRER RECHARGER..ENFIN TOUT!.

SPIDER INVASION 10 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
L'ENNEMI ATTAQUE EN RANGEE DE PLUS EN PLUS PRESSANTE.ATTENTION!!!.

SQUASH 5 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
TRES BON ENTRAINEMENT POUR BLOCK KUZUSHI BREAK-OUT ETC ETC....

STAR DESTROYER 12 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
VOUS PILOTEZ UN AVION ET VOUS ETES JUSTE DERRIERE LA LIGNE DE MIRE.

STAR FIRE 15 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
RESSEMBLE PAR LE COTE LIGNE DE MIRE A STAR DESTROYER.

SUPER DEFENDER 19 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
A MOINS D'ETRE UN SPECIALISTE VOTRE AVION EST VITE DETRUIT.

SUPER ELECTIONS 32 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
VOUS ETES EN ANGLETERRE ET IL FAUT FAIRE ELIR VOTRE PARTI.(DUR DUR!!)

SUPER HEROS 9 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
L'ANCETRE DES JEUX BASIC.LE POSSEDER MAINTENANT EST UNE VALEUR SURE!!.

SUPER WURM 7 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
AU DEBUT LA QUEUE DU SERPENT EST COURTE MAIS VOILA ELLE S'ALLONGE !.

SUPER-ANALOGUHR 9 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
IL SUFFIT DE METTRE CETTE PENDULE A AIGUILLES A L'HEURE.

SUPERHIRN AKTIV 15 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
TROUVEZ UN NOMBRE SECRET SOUS LE PRINCIPE DU MASTERMIND.PARFOIS DUR!!

TAPEWORM 13 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
UN SERPENT SE PROMENE SI VOUS TIREZ DEDANS IL SE COUPE EN DEUX.

TELECRAN 3 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
PRESQUE TOUS LES DESSINS POSSIBLE SUR L'ECRAN.BIEN POUR LES ENFANTS.

TOUR INFERNAL 12 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
VOUS ETES AU MILIEU D'UNE TOUR EN FEU !VOUS SAUVER OU MOURIR !.

TURTLE GAME 10 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
SAUTER DE PIERRE EN PIERRE AU MILIEU DE LA RIVIERE ATTENTION A LA CHUTE.

U-BOOTJAGD 6 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
VOUS PILOTEZ UN BATEAU LANCE-TORPILLES.....

VIDEO-FLIPPER 10 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
PEUT-ETRE UNE PARTIE GRATUITE SUR CE FLIPPER.MAIS PAS DE TILT!!.

VIER-GEWINNT 7 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
MORPION QUI SE JOUE A L'HONRIZONTAL.

VIERGE 7 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
IL FAUT QUE VOUS POSSEDIEZ UNE IMPRIMANTE POUR ADMIRER CETTE VIERGE.

WURM 6 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A ■ 5510 □ 5060A □ LM ■
VOUS DIRIGEZ UN SERPENT DANS TOUS LES SENS ET ENGRANGEZ DES POINTS.

LOGITHEQUE CLUB SHARP MZ 80 K/A

PROGRAMMES UTILITAIRES

COPIE TOUT 2 Ko. ▮K ■ 6015 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
DUPLIQUE RAPIDEMENT A PARTIR D'UNE DISQUETTE UNE AUTRE DISQUETTE.

COPY K7 1 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
DUPLIQUE A PARTIR D'UNE CASSETTE UNE AUTRE CASSETTE EN L.M.

COPY SINGLE 4 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
TOUTE LES POSSIBILITES DE TRAVAILLER LES CASSETTES.

CREGEFI CAS. 16 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
SUPER FICHIER PARAMETRABLE (CASSETTE).

CREGEFI DISK. 3 Ko. ▮K ■ 7015 □ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
SUPER FICHIER PARAMETRABLE (DISQUETTE).

DESASS.EDIT. 6 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
L'EDITEUR EN PLUS

DESASSEMBLEUR 15 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
COMME SON NOM L'INDIQUE.

DUMP DISK 6015 6 Ko. ▮K ■ 6015 □ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
VOUS POURREZ EXPLORER VOS DISQUETTES.

PRINTER-DUMP 0.5 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
SI VOUS POSSEDEZ UNE IMPRIMANTE.

SPONTEX 8 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
TRAITEMENT DE TEXTE(POUR L'INSTANT SANS NOTICE).

SUPER BASIC 2 20 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
QUI NE LE POSSEDE PAS ?.SUPER SUPER BASIC AVEC NOTICE.

SUPER MONITEUR 4 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
UTILITAIRE D'AIDE A LA PROGRAMMATION.

UTILIDISK 6 Ko. ▮K ■ 5025 □ 5060 □ LM ■ * A □ 5510 □ 5060A □ LM □
TRES TRES UTILE POUR CEUX QUI UTILISENT SOUVENT LES DISQUETTES.

LOGITHEQUE CLUB SHARP MZ 80 K/A

PROGRAMMES ENSEIGNEMENT

2CONDSERIE1CONNU 0.5 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
CALCUL DE CONDENSATEUR DANS UN CIRCUIT.

2RESIST.//1CONNU 0.5 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
CALCUL DE RESISTANCE DANS UN CIRCUIT.

ANGLAIS-ALLEMAND 13 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
MINI-DICTIONNAIRE.

ARITHMETIQ. 19 Ko. ▮K □ 5025 □ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
LES QUATRE OPERATIONS AU PROGRAMME.(POUR ENFANTS).

BASIC TUTO. 23 Ko. ▮K □ 5025 □ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
INITIATION A LA PROGRAMMATION BASIC.

CALC POT FERRITE 1 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
CALCULER VOS POTS FERRITE.

CONDENSAT SERIE 1 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
CALCUL D'UN CONDENSATEUR DANS UN CIRCUIT SERIE.

CONJUGAISON 14 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
ENTREZ UN VERBE ET VOTRE MZ LE CONJUGUERA A PLUSIEURS TEMPS.

CONV. ALGEBRI. 6 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
PROGRAMME DE CONVERSION ALGEBRIQUES.

E.A.O. BETON 22 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
TRES BON PROGRAMME POUR S'INITIER A LA CONFECTION DU BETON.

FRANCE 15 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A ■ 5510 □ 5060A ■ LM □
APPRENEZ LES REGIONS LES DEPARTEMENTS LES PREFECTURES.

GENEDIV. 3 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
IMPRIMANTE NECCESSAIRE.IMPRIME SUR PAPIER DES DIVISIONS(POUR ENFANTS).

GEOGRAPHIE 16 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
QUESTIONS SUR LA GEOGRAPHIE MONDIALE.

INTEGRALE 2 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
CALCUL DES INTEGRALES.

LATIN 4 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
INITIATION AU LATIN (DECLINAISONS).

MULTIPLICATION 2 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
APPRENEZ LES MULTIPLICATIONS A VOTRE ENFANT.

ORTHOGRAPHE 5 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
APPRENEZ L'ORTHOGRAPHE A VOTRE ENFANT.

PROBABILITES 20 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A ■ 5510 □ 5060A ■ LM □
TRES BON PROGRAMME AVEC DES GRAPH. A L'APPUI.(CLASSE TERMINALE).

R.L.C 2 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
CALCULEZ UN CIRCUIT RESONNANT.

RECITATION 13 Ko. ▮K ■ 5025 □ 5060 ■ LM □ * A □ 5510 □ 5060A □ LM □
PERMET A L'ENFANT D'APPRENDRE UN TEXTE.

RESISTANCES // 1 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
CALCUL DE RESISTANCES DANS UN CIRCUIT PARALLELE.

RESOLUTION 123° 2 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
CALCUL DES EQUATIONS A UNE DEUX TROIS INCONNUES.

RESOLUTION 3EQ 2 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
CALCUL DES EQUATIONS A 3 INCONNUES.

SIMU. H.P. 11 Ko. ▮K □ 5025 □ 5060 □ LM □ * A ■ 5510 ■ 5060A □ LM □
TRANSFORMEZ VOTRE MZ EN CALCULATRICE (NOTATION POLONAISE INVERSEE).

STAT. 3 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
FAITES DES STATISTIQUES.

SURFACE 0.5 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
LES SURFACES DES CARRES RECTANGLES ETC...

THE TEACHER 13 Ko. ▮K ■ 5025 ■ 5060 □ LM □ * A □ 5510 □ 5060A □ LM □
INITIATION A L'ANGLAIS.

LE CLUB DES SHARPENTIERS

POURQUOI ?

Le monde de la micro-informatique évolue à une vitesse prodigieuse. Cela nécessite, pour le possesseur d'un micro-ordinateur, de se tenir constamment informé de toute évolution ou création, tant matérielle que logicielle réalisée autour de sa machine. Cette nouvelle forme de « service après vente », c'est l'objectif n° 1 du **CLUB DES SHARPENTIERS**. Né en 1980 du désir de quelques passionnés de partager leurs connaissances, le club a acquis, en quelques années, une solide expérience en ce domaine.
Cette expérience, l'inconstestable succès rencontré – actuellement + de 7 000 Sharpentiers –, la qualité et la diversité des services proposés font du **CLUB DES SHARPENTIERS**, une organisation actuellement unique en France.

QUELS AVANTAGES ?

Etre SHARPENTIER, c'est d'abord adhérer à la philosophie du **CLUB DES SHARPENTIERS** qui, au-delà de toute considération commerciale, a pour seule tâche de favoriser la communication entre possesseurs de micro-ordinateurs SHARP.
Cette communication, c'est d'abord le **bulletin du club**, bimestriel,qui vous fournit les informations concernant votre machine (applications, nouveautés, programmes), la vie du club et de la Société SHARP.

La vie du club, c'est aussi une **permanence hebdomadaire** le mercredi dans les locaux de SHARP à Aubervilliers où les Sharpentiers peuvent se réunir et travailler dans un « local club ».
Par ailleurs, le club propose une **assistance permanente**, téléphonique ou postale, des membres du club qui mettront tout au œuvre pour résoudre votre problème ; l'accès à la plus importante **bibliothèque de programmes** ; un échange constant d'idées permettant à chacun de profiter du savoir de tous, et à tous, de l'acquis de chacun.
Le club étant parrainé par la Société S.B.M., il convient d'ajouter à cette liste, la possibilité d'obtenir certains produits SHARP à des prix très intéressants (affaires club) et de pouvoir participer à des concours club dont les prix sont de grande valeur (1 semaine au japon au vainqueur 1985 de notre concours de DAMES).

POUR DEVENIR SHARPENTIER...

Remplissez le bulletin d'inscription ci-joint et retournez le au :
Club des Sharpentiers
151/153, avenue Jean-Jaurès
93307 AUBERVILLIERS CEDEX
Tél : 834.93.44

BULLETIN D'INSCRIPTION AU CLUB DES SHARPENTIERS

☐ Je m'inscris
☐ Je me réinscris
au CLUB DES SHARPENTIERS

Je bénéficie de tous les avantages du CLUB

Je suis abonné pour 1 AN au BULLETIN du CLUB

Je vous joins mon règlement
☐ FRANCE : 160 F
☐ ETRANGER : 200 F

CHEQUE N° .... BANQUE ......

DATE SIGNATURE

n° 10

NOM .... PRÉNOM ....

ADRESSE ....

CODE POSTAL / VILLE ....

PAYS ....

PROFESSION .... ÂGE ....

MACHINE POSSEDÉE .... DEPUIS ....

ACHETÉE CHEZ ....

UTILISATION PRINCIPALE DE VOTRE MACHINE ....

....